REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
01-11-2016	Annual Research Report	20151001 - 20160930

4. TITLE AND SUBTITLE

Micro Autonomous Systems Research
Systems Engineering Processes for Micro-Autonomous Systems

5a. CONTRACT NUMBER LaRC/ARL Coop NNL09AA0A

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

K. Daniel Cooksey, Minki Kim, Devan Thanki, Braven Leung, Zach Fisher, Arthur Cheng, Raphael Gautier, Coline Ramee, David Locascio, Eric Schaus, Dimitri Mavris

5d. PROJECT NUMBER 2B34

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Aerospace Systems Design Laboratory, Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0150

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

US Army Research Laboratory
RDRL-VTV
6340 Rodman Rd
Aberdeen Proving Ground, MD 21005

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

This material is based on work supported by the National Aeronautics and Space Administration, Langley Research Center under the Research Cooperative Agreement No. NNL09AA00A awarded to the National Institute of Aerospace. Any opinions, findings,

14. ABSTRACT

In Phase III of the Micro-Autonomous Systems Research (MASR) project, the Georgia Tech ARL team extended an automated product family engineering process and toolset allowing the creation of tailored one-off unmanned aerial System (UAS) solutions to Soldier needs. The toolset provides a simplified user interface for non-technical users to enter vehicle requirements, such as sensor packages, endurance, payload, etc. A logistics interface allows an untrained logistics operator to enter machines and parts availability. This information is fed to a set of engineering analyses where a feasible design (if possible) is generated, and the drawings for manufacture are output. These part designs are then provided to a technician with automated manufacturing tools (such as 3-D printing) who starts the automated manufacturing, assembles components, and delivers the tailored UAS to the Soldier. This process has been validated via flight tested vehicles and satisfies the desire to be more responsive to Soldier needs for small UASs.

15. SUBJECT TERMS

unmanned systems, design space exploration, systems engineering, integrated design and manufacturing, additive manufacturing, 3D printing, on-demand, forward deployed, conceptual design, sizing, architecture, experimentation, scenario simulation, technology tradeoffs, fixed wing, multicopter, quad copter

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
				43	Eric Spero
					19b. TELEPHONE NUMBER *(Include area code)* 410-278-8743

Reset

MICRO AUTONOMOUS SYSTEMS RESEARCH

GEORGIA INSTITUTE OF TECHNOLOGY

DR. K. DANIEL COOKSEY
MINKI KIM
DEVAN THANKI
BRAVEN LEUNG
ZACH FISHER
ARTHUR CHENG
RAPHAEL GAUTIER
COLINE RAMEE
DAVID LOCASCIO
ERIC SCHAUS
DR. DIMITRI MAVRIS (PI)

NOVEMBER 1, 2016

INTRODUCTION

In Phase III of the Micro-Autonomous Systems Research (MASR) project, the Georgia Tech ARL team extended an automated product family engineering process and toolset allowing the creation of tailored one-off unmanned aerial System (UAS) solutions to Soldier needs. The toolset provides a simplified user interface for non-technical users to enter vehicle requirements, such as sensor packages, endurance, payload, etc. A logistics interface allows an untrained logistics operator to enter machines and parts availability. This information is fed to a set of engineering analyses where a feasible design (if possible) is generated, and the drawings for manufacture are output. These part designs are then provided to a technician with automated manufacturing tools (such as 3-D printing) who starts the automated manufacturing, assembles components, and delivers the tailored UAS to the Soldier. This process has been validated via flight tested vehicles and satisfies the desire to be more responsive to Soldier needs for small UASs.

PROJECT HISTORY

A vital requirement of the modern combat environment is to gain and maintain situational awareness to facilitate effective squad-level decision making. Over previous years, Georgia Institute of Technology (Georgia Tech) has collaborated with the Army Research Laboratory (ARL) in developing design capabilities to assess the operational capability of micro autonomous vehicles[1] to assist at the squad level. Improved systems engineering processes for micro-autonomous systems is the primary focus of the research undertaken in the Micro-Autonomous Systems Research MASR effort. This report details the work completed in Phase II of the joint effort between ARL and Georgia Tech.

Phase I was focused on research and development of a systems engineering process to design and flight test an autonomous aerial system for use within a building. Emphasis was placed on the development and application of a systems engineering process, which resulted in a flying prototype capable of mapping the interior of a room.

In Phase II, the research was focused on how to accelerate the systems engineering process for rapid response to changing Soldier needs. Acceleration of the systems engineering process was achieved by integrating modeling, design, and manufacturing tools and incorporating extensive use of modularization.

In Phase III, the MASR project succeeding in delivering an engineering methodology for the enabling of On-Demand Small Unmanned Aerial Systems (ODSUAS). This methodology allows an air vehicle to be algorithmically designed in response to a soldier's needs. Once the soldier has input their mission needs, the methodology generates a feasible

[1] The term "micro autonomous vehicles" refers to Soldier-borne aerial sensors that range in size from hand-held to approximately 24 inches. As of the writing of this report, a formal definition of "micro autonomous vehicles" does not exist within the US Armed Services; the smallest recognized UAS Group encompasses 0 to 20 lbs take-off weight.

design of a small unmanned aerial system SUAS. This SUAS can be constructed in-situ from algorithmically generated parts which are printed on a 3D printer, and algorithmically selected subsystems (such as autopilots and motor control circuits) which have been identified to be in the current inventory of parts.

REPORTS TRUCTURE

The following report is a compilation of three conference papers detailing the various SUAS vehicles developed during the effort, as well as a paper documenting in detail the process for ODSUAS design and development. The following papers have been publicly vetted, and the work contained within these papers is to be tested at the Army Expeditionary Warfighting Experiment 2016 in November of 2016.

The first paper was presented at the ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference in Charlotte, north Carolina in August 2016. The next paper was presented at the AHS 72^{nd} Annual Forum in West Palm Beach, Florida in May 2016. Finally, the third paper was presented at the AIAA Aviation Forum 16^{th} Aviation Technology Integration and Operations Conference in Washington, District of Columbia in June 2016.

ADAPT DESIGN: A METHODOLOGY FOR ENABLING MODULAR DESIGN FOR MISSION SPECIFIC SUAS

Zachary C. Fisher
David Locascio
K. Daniel Cooksey
Dimitri N. Mavris
Georgia Institute of Technology
Atlanta, GA, USA

Eric Spero
US Army Research Laboratory
Aberdeen, MD, USA

ABSTRACT

Recent advances in small unmanned aircraft systems (SUAS) have greatly broadened the scope of their potential applications. However, traditional design processes applied to SUAS produce a single design for a single set of requirements. Off-design mission performance is often greatly degraded due to the vehicle's small scale. This paper considers a different approach to SUAS design aimed at addressing this issue. In this approach, a hybrid modular and scalable product family is coupled with linked engineering analyses in order to automatically formulate a design given a set of mission requirements. This allows multiple SUAS designs to be rapidly synthesized from multiple sets of design requirements using a common set of components. Designs are then rapidly generated and manufactured "on-demand" using automated manufacturing techniques in order to address unforeseen mission needs.

The design approach, named "**A**ggregate **D**erivative **A**pproach to **P**roduc**t** **Design**" (ADAPt Design), consists of four actions: (1) requirements analysis, (2) architecture selection, (3) interface design, and (4) concept refinement and design. The outcomes of the method are a family of designs which are highly compatible with design automation, and a toolset that automatically translates changes in requirements to changes in detailed 3-D models. Results of the application of this approach are presented via the design of several SUAS. The capability of the design paradigm is assessed through a comparison of design requirements to the measured performance of the designed vehicle, and conclusions are drawn about the approach's applicability and scalability.

INTRODUCTION

Presently, U.S. Army UAS are primarily used to support tactical operations through collection of intelligence, surveillance, and reconnaissance (ISR) information. Ideally, troops in the field would employ UAS assets on-demand to acquire Actionable Intelligence in real-time.

An assessment of U.S. military operations in the suburbs of Baghdad, Iraq conducted by the RAND Corporation for the U.S. Army concluded that modern combat operations increasingly require decentralized decision making. It states

> "The enemy is fleeting, which means that decentralized decision making is required. Units at the brigade level and below must therefore have access to the information and other capabilities required to support the rapid decisions necessary to deal with a highly mobile enemy … and to enable effective, independent action [1]."

The U.S. Army unmanned aircraft systems roadmap for 2010- 2035 supports this conclusion. Furthermore, it recommends that UAS be used to enable decentralized decision making. It states

> "UAS require and enable accelerated multi-echelon, decentralized decision-making, and execution, significantly changing the tempo and dynamics of operations. Lower echelon leadership must be empowered with authority and bandwidth to employ UAS as their changing situation dictates, operating at a tempo that is faster than higher echelon leadership can affect. [2]"

The U.S. Army Training and Doctrine Command (TRADOC) recognizes that the modern battlespace is a rapidly evolving environment that demands responsiveness from Soldiers and their equipment to maintain dominance [3]. SUAS provide a means to develop situational understanding in support of decentralized decision making during future expeditionary operations envisaged by TRADOC.

Accordingly, SUAS have increasingly been used to provide battlefield situational awareness. SUAS can perform functions such as intelligence, surveillance, and reconnaissance (ISR), security, manned-unmanned teaming (MUM-T), communications relay [4], finding IEDs, identifying enemy combatants [5], and performing advance scouting all with greatly reduced risk to the soldier [4]. The vehicles currently in use by the U.S. Army can be broken into three categories – division level and above, brigade level, and battalion level and below [4]. Oftentimes, it is difficult for personnel at the battalion level and below to procure and use SUAS assets due to the limited quantity of vehicles available to the Army. Many of these vehicles are also limited in the missions that they can perform due to having been designed around a specific set of requirements.

Equipping a Solider with a SUAS can take one of three approaches:

1) **Multi-mission asset:** One SUAS asset covers all mission needs while sacrificing performance across all missions
2) **Set of optimized assets:** A set of SUAS assets designed for a subset of specific missions are deployed; troops may need to carry a large number of assets to cover all possible missions
3) **Asset on-demand:** One-off asset is specifically tailored to and custom manufactured for the mission it will perform

These approaches are depicted in Figure 1 for three notional performance metrics. The square represents a multi-mission asset designed to operate in the center of the capability space. The black dots show a set of optimized assets occupying discrete points in a slightly wider space. The gray dots show designs that are generated on-demand and tailored to the need at hand. Figure 1 illustrates that an on-demand approach captures the best of the multi-mission and optimized assets approaches: it can cover a diverse range of mission needs without imposing a logistical burden on the Solider of having to carry a portfolio of assets.

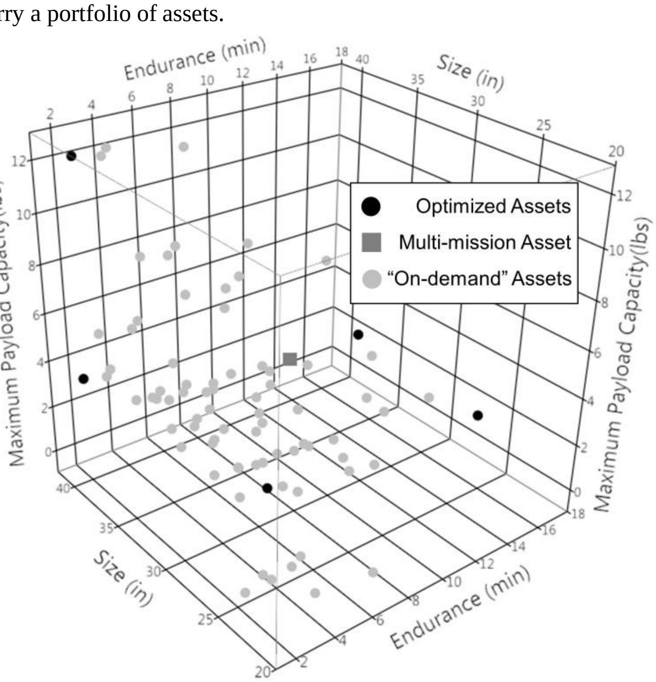


FIGURE 1: THREE APPROACHES TO SUAS DEVELOPMENT ADDRESSING DIVERSE MISSION NEEDS

BACKGROUND AND LITERATURE REVIEW

The design methodology presented in this paper couples concepts from product family design and reconfigurable system design with recent developments in the fields of automated manufacturing and micro-autonomous systems. A brief overview of these topics and relevant research efforts are described in this section to give context to, and establish a consistent lexicon for the work presented in this paper.

Automated Manufacturing

An on-demand approach requires decentralized decision making power and access to automated manufacturing capabilities, as well as supporting doctrine and processes. Current acquisition and requisition methods are incompatible with this vision. Materiel procurement cycles are generally long, requiring an identification of the need and establishment of formal requirements using the Joint Capabilities Integration and Development System (JCIDS), and translation of those requirements into materiel solutions using the process outlined in the Defense Acquisition System. Requisitioning supplies is less arduous, but still requires a formal approval process. New manufacturing techniques such as 3-D printing, consumer-focused computer numerical control (CNC) milling and laser cutting enable rapid manufacturing of one-off systems and parts. These techniques offer the potential for invention, innovation, modification, and manufacture to be forward deployed at the point of need and are an enabler for the on-demand vision of allowing Soldiers to create materiel prototype solutions to unforeseen or unanticipated problems. These manufacturing techniques are already accessible in some capacity as part of the U.S. Army Rapid Equipping Force (REF) Expeditionary Labs (Ex Labs), a team of trained personnel equipped with mobile manufacturing equipment [6].

Micro-Autonomous Systems Research (MASR)

Georgia Institute of Technology has collaborated with the U.S. Army Research Laboratory (ARL) in researching capabilities for assessing the operational utility of small autonomous systems assisting at the squad level. Improved systems engineering processes for these systems is the primary focus of the research undertaken in the multi-year MASR effort.

Previous work has explored a multidisciplinary framework built on the simultaneous application of decomposition and re-composition approaches, and was implemented to provide a structured, traceable method for evaluating mission effectiveness of systems of microsystems [7]. This culminated with an Interactive Reconfigurable Matrix of Alternatives, a tool for comprehending the large concept solution space. Fundamental mission requirements included endurance, adaptability, path planning, and communications [8].

Ref. [9], entitled "An Automated Approach to the Design of Small Aerial Systems Using Rapid Manufacturing", explores development of the systems engineering processes necessary for the development and test of an autonomous system for use within a building's interior. The work presented in this paper is a direct continuation of the developments in Ref. [9], with a focus on outdoor aerial operations.

Product Family and Product Platform Design

A *product family* is a group of similar products derived from a common platform. Individual products belonging to a family are called *variants*, and each variant has a set of distinguishing features which allow it to meet different requirements than other variants [10],[11]. The advantages of using product families to derive new designs stem from the reuse of major design elements. A generic development process for product families is presented by Jiao et al. Development starts by defining a set of product functional requirements that address the defined customer needs. Next, functional requirements are mapped to design parameters. These mappings are the mechanism by which physical product designs are formulated. Finally, manufacture of the product variants is coordinated by mappings between design parameters and process variables. In this final stage, consideration is given to sharing manufacturing processes and supply chain logistics across variants [10].

Product families have been the subject of extensive research, categorized into several key issues by Pirmoradi et al. [12]. Of specific importance to the design approach presented in this paper are the issues of product architecture, platforms, variety versus commonality, and design optimization. *Product architecture* refers to relationships between a product's components and the mappings between functional requirements and individual components in the product. *Platforms* are

the groupings of "components, technologies, subsystems, processes, and interfaces" that form the basis from which variants in a family are derived [13]. Two types of platforms have been identified: *scalable platforms* where variants are produced by varying scalable design variables, and *modular platforms* where variants are produced through the exchange of different modules. A characteristic of modular platforms is a one-to-one mapping between functional element and physical components [14]. *Variety versus commonality* refers to the tradeoff between retaining common features between variants and ability of the product family to meet a wide range of customer needs. Finally, *design optimization* refers to the set of techniques used to determine values of the design variables which result in a design that best meets objectives established from the customer needs [12].

An important semantic distinction is the one between product architecture and product configuration. *Product architecture* refers to the arrangement of functional elements into physical units and the interaction between these units [15]. *Product configuration* refers to the spatial layout of physical components, features, and modules. In the context of a product family, configuration defines the allocation of these elements between product variants [12].

This work borrows concepts from product family design to enable design automation of SUAS. Deriving variants from a fixed, common product platform separates configuration development, which is difficult to automate, from preliminary and detailed design activities which are more readily automated. This directly leads to the possibility of in-situ SUAS development where vehicle designs are tailored to a wide range of requirements.

Reconfigurable System Design

Reconfigurable systems are systems that can reversibly take on distinct configurations through alteration of form or function in order to achieve differing levels of system performance [16],[17]. Often, reconfigurablity is employed to permit systems to operate closer to their optimal performance under a wide range of operating conditions by trading between competing performance metrics [16]. The topic of reconfigurable systems was first introduced as a topic of product design research by Olewnik, Brauen, Ferguson, and Lewis, who describe methods to characterize such systems and assess the flexibility they permit during a design process [17]. Literman, Cormier, and Lewis further present a framework to fully characterize reconfigurable design concepts, which require additional information over their static counterparts. Such a characterization framework is needed to compare between concept alternatives [18]. Of particular relevance to this paper are the developments of Patterson, Pate, and German, who consider a UAS which is built from modular airframe components that can be interchanged between flights. The authors demonstrate several approaches to assess the flexibility in UAS performance attained through reconfiguration of the vehicle [19].

The on-demand design philosophy described in this paper exists at the junction of product family design and reconfigurable system design. More specifically, the design philosophy achieves adaptability not by physical reconfiguration of an individual product, but rather through on-the-fly reconfiguration of the product's design. This notion results in a set of designs that in many ways resembles both a product family and a reconfigurable product.

RESEARCH OBJECTIVES

The on-demand approach is succinctly explained via an analogy to Lego® depicted in Figure 2. Lego® bricks contain a number of modular parts that can be constructed into different models depending on what outcome is desired. Instructions are provided to help the user build different systems out of the same set of components. In the context of this work, a small set of off-the-shelf parts which cannot currently be manufactured on site, such as motors, propellers, and control electronics, will be provided ahead of time to a supply facility at a forward operating base. These off-the-shelf parts will then be combined with parts manufactured on-site to create the needed system.

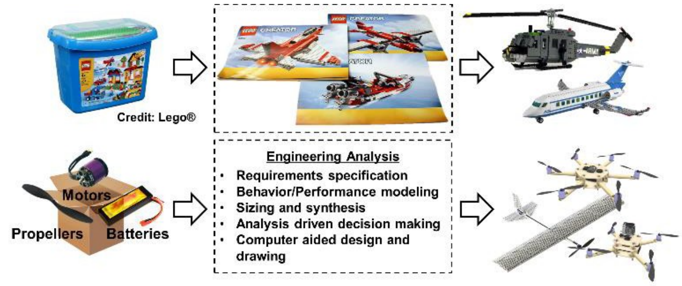


FIGURE 2: DESIGN ON-DEMAND ILLUSTRATED VIA ANALOGY TO LEGO®

Figure 3 depicts a potential concept of operations for implementing the on-demand approach to an immediate Soldier need. In this case, the underside of a bridge needs to be inspected using a hover-capable system. The patrol relays their mission needs to a supply facility (e.g., forward operating base or REF Ex Lab), where an integrated engineering workflow is used to design a SUAS tailored to the immediate mission need. Within hours, the solution system is delivered to the Soldiers who use it to inspect the bridge. Alternatively, prior to conducting a patrol, the integrated engineering workflow can be used to create alternate SUAS as individual units of issue. Soldiers would then requisition and pick up the systems from the supply facility prior to departing on the patrol.

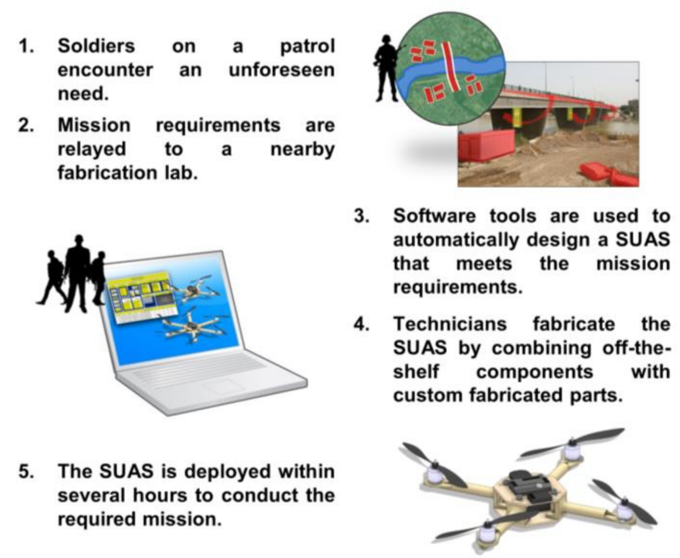


FIGURE 3: CONCEPT OF OPERATIONS.

Several challenges must be addressed before an on-demand approach can be realized. First, technicians must be trained to use the integrated engineering workflow as well as the available automated manufacturing equipment. Manufacturing technicians are envisioned to be involved in this process primarily because the Soldier's focus will always be on the mission, but also because the process aims to eliminate the need for a high level of engineering background.

Another challenge is that the design process must be developed in such a way that the vehicles perform as they are intended to. The

implication is that each SUAS designed and developed using the integrated engineering workflow will not be tested before operational use. The intent is to be able to move directly from design to manufacture and then to deployment. This challenge is perhaps best illustrated by the systems engineering "vee" model in Figure 4 which captures the systems engineering actions of a general development cycle. The left leg translates customer requirements into system requirements, and then decomposes the design into increasingly specific subsystems. At each step, a plan is established to verify and validate the resulting subsystems by following the right side of the "vee". Traditional design processes rely on having an assembled product to conduct verification and validation. This is not the case in an on-demand approach, where the entire re-composition (e.g., verification and validation) leg of the "vee" will have to be collapsed into the decomposition leg. Our approach for developing trusted on-demand systems relies on pre-validating platform architectures, configurations, components, and subsystems, and leverages computer based modeling tools and automated manufacturing techniques.

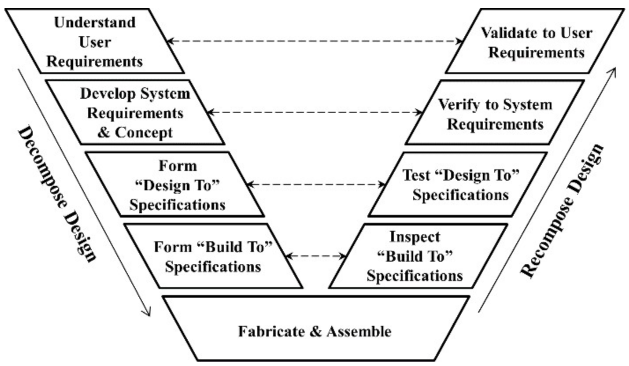


FIGURE 4. FORSBERG AND MOOZ SYSTEMS ENGINEERING "VEE" MODEL [20]

This paper presents a method to define and architect a set of product platforms that are highly compatible with design automation. In this context, the set of platforms is best described as a product family, except that the product variants exist not as discrete entities but as potential designs that vary continuously over a fixed design space. This is a departure from the traditional concept of a product family which is only concerned with a finite number of designs occupying discrete parts of the design space.

The method developed in this work has been named "**A**ggregate **D**erivative **A**pproach to **P**roduc**t** **Design**" (ADAPt Design). New designs are derived from aggregations of pre-determined components and design elements. The platforms are a hybrid modular and scalable architecture. Some components can be swapped one-for-one to form a new variant, while others have features that vary continuously. This notion is illustrated in Figure 5.

At its core, ADAPt Design uses rigorous systems engineering techniques to form an executable link between input requirements and an output design. "Executable" is not intended to take on an abstract meaning but instead indicates that the aforementioned link is documented in executable code. The code takes requirements as inputs and uses them to directly drive computer aided engineering and design (CAE/CAD) tools which output detailed models and manufacturing files.

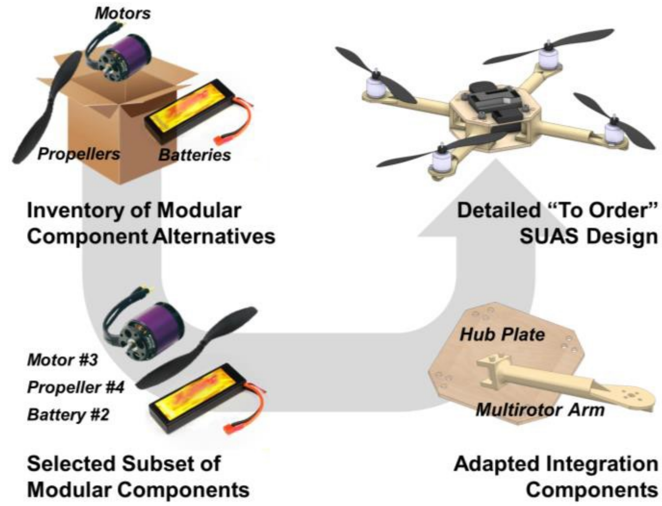


FIGURE 5. ADAPT DESIGN CONCEPT ILLUSTRATION

ADAPT DESIGN METHOD

ADAPt Design is presented in this paper as a linear process divided into four actions: (1) requirements analysis, (2) architecture selection, (3) interface design, and (4) concept refinement and design. It is important to note that in reality, the methodology is like all design processes in that implementation is iterative in nature and occurs over time.

1) Requirements Analysis

ADAPt Design begins by determining and documenting the range of needs to be addressed by the product family. By the conclusion of requirements analysis, the following should be identified:

1) Broad definitions of objectives to be fulfilled by the product family in terms of capabilities. Specifically, this means articulating clearly whose needs will be addressed along with a statement pertaining to how they will be addressed.
2) Key stakeholders in the product family's use and the requirements and constraints they impose on its development. These requirements and constraints will be used to both bound and validate the product family architecture.
3) Engineering metrics against which derived designs will be measured and compared. These are typically quantifiable characteristics of each design and may include performance metrics, physical dimensions, and required manufacturing time.

2) Architecture Selection

The goal of architecture selection is to identify and define the product platform(s) that will comprise the product family by using the using the objectives, capabilities, requirements, constraints, and metrics established during requirements analysis. All variant designs will be derived from one of the platforms, and so at least one platform should be identified to cover each of the established capabilities. The number of potential platforms is generally exceedingly large, and so a systematic approach to identifying the most promising platforms is needed. Three approaches are (1) a functional decomposition, (2) a historical search, and (3) requirements space coverage.

In the functional decomposition approach, all of the functions required to achieve the specified capabilities are listed. Components are then matched to each function to establish a list of components that

fulfill all functions. This list is then used to build platforms via a morphological matrix.

A historical search can be used to identify classes of products that have previously been used to achieve the capabilities of interest. The result of this exercise is a list of potential platforms and an understanding of the gaps in capabilities which remain.

The requirements space coverage approach focuses on the capabilities gaps. Here, individual platform concepts are conceived in an attempt to sufficiently eliminate coverage gaps. Each concept is analyzed to understand the capability gap it fills and which capabilities remain unaddressed.

A concurrent application of each of these three approaches is recommended to identify the set of platforms that best meets the established capabilities. Consideration must be given to the tradeoff between variety versus commonality. Increasing the number of platforms covers more requirements at the cost of reduced platform commonality. This in turn increases design overhead and logistics related to procuring and holding parts in inventory.

The next major step in architecture selection is a functional decomposition of the selected platforms, and a subsequent allocation of subsystems and individual components to each function. The resulting list of components and subsystems is then inspected to identify which components and subsystems are common across platforms and which are platform-specific.

Components are further classified as being "modular" or "scalable." A modular component indicates that generating variant designs is achieved by swapping discrete alternatives of that type of component. Modular components will be supplied to the user beforehand and will be used with little or no modification. Scalable components can scale via a limited set of continuous design parameters. Examples include part dimensions, instances of geometric patterns, and locations of features. This classification is not exclusive; components can have nested modular and scalable elements. For example, a beam feature is fabricated by cutting a length of an aluminum tube. Two alternative tubes are supplied: a circular cross section and a square cross section. In this case, the beam's cross section is modular while its length is scalable.

To document the design decisions made to this point, a formal organization of components should now be built in the form of a component library. The component library enumerates key information related to each component or subsystem. This includes its functional role, its classification as modular or scalable, its scalable variables if applicable, any interfaces, and all engineering data that is pertinent to its use in a design or modeling its performance. Engineering data is subsequently referred to as "attributes." The component library will serve as the primary source of information regarding available component alternatives as required by automated engineering analyses.

Architecture selection concludes with a step akin to traditional conceptual design. If not already completed as a byproduct of platform identification, a preliminary definition of each platform's layout should now be established. The result is a set of configuration layouts for each platform that show rough component layouts and bounds on the interfaces between subsystems. Additionally, the scale of the modular components should be identified. For example, a rough SUAS sizing exercise would indicate the range of electric motor sizes that will be of interest.

Of critical importance is that this step abandons the traditional notion that "major design changes are frozen at the end of conceptual design." The intent of architecture selection is to identify platforms that are highly adaptable and so the conventional notion is counterproductive. Instead, the configuration layouts should indicate which components and requirements will drive interfaces, the magnitude of variation expected for the interfaces, and a rough idea of the location of components, subsystems, and interfaces.

It is recommended that the configuration layouts are documented in the form of "model skeletons," which are 3-D representations of key geometric planes, points and shapes located in space. Implemented in a CAD environment, the model skeleton approach has been branded "top down design" by some in the community. A model skeleton and the quadrotor SUAS it represents are depicted in Figure 6.

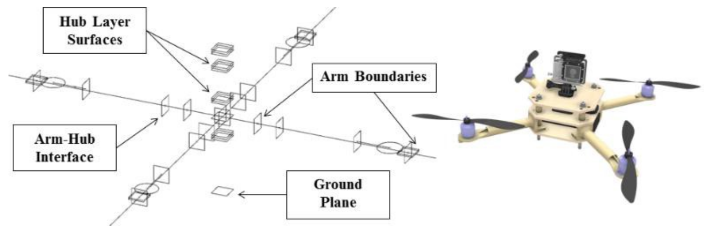


FIGURE 6. 3-D DEPICTION OF A MODEL SKELETON AND THE REPRESENTATIVE QUADROTOR SUAS

Model skeletons serve two purposes. The first is to aid the designer in capturing the configuration layout of the platform. These models will be heavily refined as development progresses, but explicitly stating key divisions helps to organize the process. The second is that it helps to remove iterative update cycles during 3-D modeling parts of the design process. During the detailed design stages, individual parts will reference the model skeletons for shared geometric features. Automated operations working on a detailed 3-D model can be very slow; operating on a model skeleton is much faster. Furthermore, resultant interface geometry is more likely to be consistent across parts. The skeleton model generated at this point does not have to contain all of the key points or elements that are likely to appear in the final design. Of more importance is capturing the elements that will be tracked as the underlying structure of the platform and how those elements relate to each other.

3) Interface Design

It was previously mentioned that freezing conceptual design changes is delayed relative to traditional design processes. While the concept is intended to remain flexible, the interfaces between flexible components must be clearly described. Interfaces can be geometric, electrical, or logical as in the case of digital communications. A novel characteristic of ADAPt Design and one that distinguishes it from more traditional design processes is an early emphasis on interfaces; locking down interface definitions provides a standardized mechanism to which new candidate variant designs will conform.

Modular components by definition are not modifiable. Therefore, the interfaces of these components are prescribed by the interface standards attached to those components and any derived variant design must adhere to those interface standards. A first step in defining interfaces is therefore identifying the interface standards of the modular components and capturing them in both the component library and model skeletons.

After interfaces of modular components have been documented, control over the remaining interfaces lies in the hands of the designer. The designer should therefore leverage the previously developed configuration layouts to define custom interface standards. These custom interface standards will be applied to all variant designs, and have additional degrees of freedom over their modular component-derived counterparts.

At the conclusion of interface design, all new geometric information generated should be captured in the model skeletons.

Locations of interfaces and parametric interface geometry are design features shared between components. As such, they are best stored in the model skeleton. Updating the model skeletons with points, planes, and representative interface geometry serves to document the interface design work in a form usable by automated design tools.

4) Concept Refinement and Design

Concept refinement and design includes many of the design activities normally associated with traditional preliminary and detailed design. In the context of ADAPt Design, elements of these activities differ in three key ways.

First, design activities are not performed manually but rather are encoded into a set of tools and then linked together. The result is a chain of linked engineering analysis tools, models, and automated decision making capabilities. The combination of these linked tools with the previously established skeleton models is the overall enabling mechanism that takes customer needs in the form of capability requirements, automatically converges to a design variant solution, and outputs a detailed set of models and manufacturing files.

The second key difference is that in traditional processes, the bills of materials and manufacturing techniques are finalized post-design, for the purposes of minimizing manufacturing risk and cost, while maximizing manufacturing throughput. ADAPt Design is meant to enable on-demand design based on automated manufacturing techniques using a common set of off-the-shelf components. As a result, all derived design variants inherently conform to pre-established logistics and manufacturing constraints.

The third distinguishing element is the importance of reducing error in all models and analyses. As described previously, the user expects to be able to assemble the design and use it immediately to meet his or her needs. Essentially, the burden of product verification and validation is transferred from the assembled subsystem or product to the models and analyses used in designing the product. Care must be taken to ensure that modeling error is acceptably low so that predictions closely match the observed behavior of the assembled product.

A primary task in concept refinement and design is to refine and supplement the constraints identified during requirements analysis. The goal is to define a complete set of constraints that the design tools will need to enforce. Constraints can be from different categories such as design, manufacturing, assembly, logistics, or regulatory, and should be quantified where possible. Early identification of constraints aids in architecting the design tools in a way that facilitates automation.

The bulk of the design automation is enabled by executable model-based design and development techniques. These techniques and the architecture of the links between them are problem specific. However, it may be beneficial to divide them into two categories: conceptual/preliminary models responsible for determining driving design parameters which control the model skeleton, and detailed design modeling which controls the lower level geometry and brings the design to a point where it can be manufactured.

Conceptual and preliminary modeling tools are responsible for determining which modular components will be used in the variant design. Candidate designs are synthesized by pairing combinations of modular component alternatives with values of high level design variables. The design variables are passed as parameters to drive updates in the model skeleton. This flow of data is depicted in Figure 7.

For simple systems composed of only a few components, a full-factorial search of component combinations in the design space is possible. For more complex systems, appropriate discrete-variable optimization techniques can be used. It is important to note that the problem is likely to be multi-objective and thus will require a multi-objective optimization routine such as NSGA-II [21] coupled with a multi-attribute decision making technique. This is the approach taken in the developments in Ref. [9].

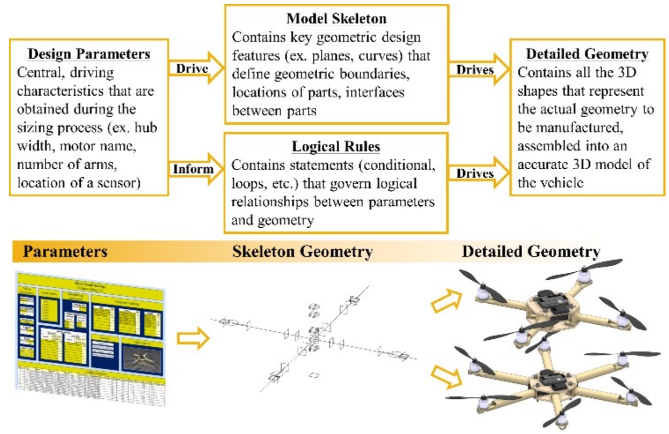


FIGURE 7. DATA FLOW BETWEEN MODEL-BASED DESIGN TOOLS

Detailed design modeling tools are responsible for translating design parameters and the chosen component alternatives into 3-D representations of parts with enough detail that they can be manufactured. Part geometry should be drawn as parametrically as possible, meaning that key dimensions are left as parameters which are referenced by lower level dimensions and geometric features. For example, the overall length of a multirotor arm is left as a parameter. This parameter is referenced by the arm tube and the motor mount pad features to size each and precisely locate their intersection.

The detailed design tools contain more than just 3-D models. They must be paired with logical rules which enforce design logic during model updates. These rules can be in the form of conditional statements, checks, or iterative loops. An example is a check on the wall thickness of a cantilever beam feature. The bending moment at the root of the beam will vary with beam length. After a change in beam length, a check rule implemented in code computes whether the thickness of the beam is enough to handle the new bending moment and increases the thickness if necessary. Logical rules are the mechanism by which impossible or invalid geometries are avoided. Several iterations of design and testing the rules will likely be necessary to achieve a working rule set. A range of 3-D modeling techniques can be used to avoid invalid geometries. A detailed exploration of this aspect of this work can be found in Ref. [22].

A concurrent design task is to identify the automated manufacturing techniques or manufacturing processes that will be used to fabricate each part. The manufacturing processes available may be subject to equipment or logistics constraints. Furthermore, each manufacturing process constrains the geometry of any part that it is used to produce. A relevant example is that 3-D printed parts with sharp corners have very high stress concentrations. Sharp edges should be filleted to prevent rapid failure of the part. On the other hand, laser cutters remove material during cutting operations, leading to a dimensional deviation from the CAD model. As such, each part must be designed with a manufacturing process in mind.

Throughout the modeling and design process, it is vital that the models for the vehicles and individual parts are validated for the range in which they will operate. For the purposes of ADAPt Design, the model must be validated across the whole expected range of the design

space. This can be accomplished by testing the extremes and a few center points.

CASE STUDIES

ADAPt Design was developed around a multirotor SUAS platform and applied to a fixed wing platform to assess its extensibility. This section illustrates ADAPt Design by following the development of these platforms starting from requirements analysis and ending in a linked set of model-based design tools.

Requirements Analysis

ADAPt Design is intended to equip Soldiers with one-off systems tailored to squad level mission needs. Stakeholders in this scenario are the Army personnel responsible for planning and executing ISR missions and the personnel manufacturing the SUAS. An assessment of the technical skill levels across these stakeholders drives a need for providing users with a small set of inputs that can fully capture the mission, without requiring detailed knowledge of design or aeronautics. Intuitive mission requirements such as payload type, range, endurance, speed, and size are chosen as inputs. To meet the on-demand vision and fill the capability gaps left by SUAS currently in the Army inventory, a need is established for converting inputs to a functional design in less than 48 hours.

Previous efforts have identified five representative mission profiles where SUAS could provide support to squad level operations: convoy surveillance and defense, perimeter surveillance and defense, building interior reconnaissance, cave interior reconnaissance, and jungle reconnaissance [8],[9] . The capabilities desired in these missions are addressed by a SUAS that carries one of four payloads: a video camera, communications equipment, LIDAR, or a target designator. From the identified missions and payload types, engineering metrics associated with performance requirements are identified in Figure 8.

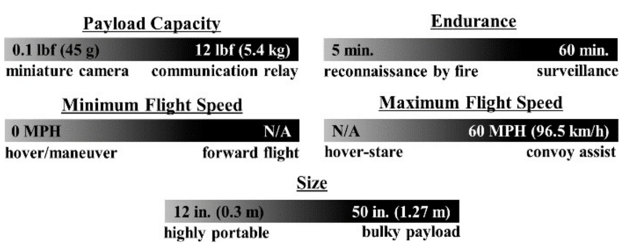


FIGURE 8. SUAS PERFORMANCE REQUIREMENT METRICS AND THEIR EXPECTED RANGE OF VALUES

Architecture Selection

A historical search approach yields two types of existing SUAS platforms that are simple, well understood, and cover all of the desired capabilities. These platforms are a multirotor for operations such as reconnaissance of building interiors or caves which require hovering and maneuvering, and a hand launched fixed wing SUAS to cover long endurance and convoy support type missions. A functional decomposition of each platform yields the functions in the left columns of each box in Figure 9.

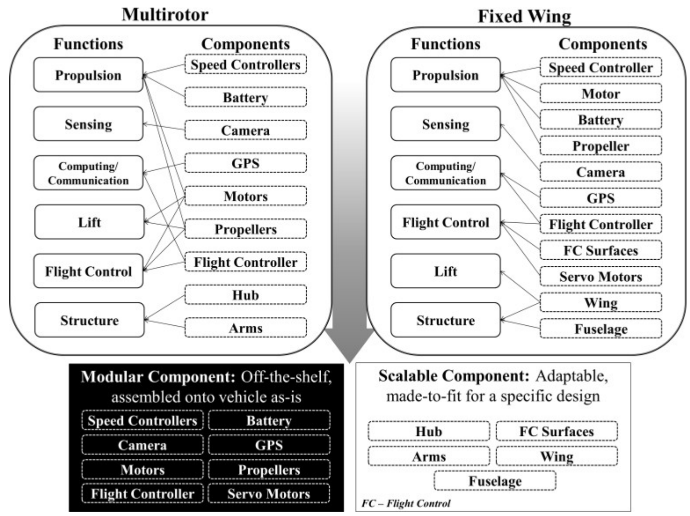
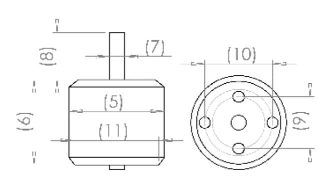


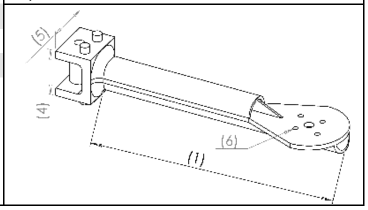
FIGURE 9. FUNCTIONAL DECOMPOSITIONS AND COMPONENT MAPPINGS OF SUAS PLATFORMS

At the bottom of Figure 9, all of the components identified to fulfill the platform functions are classified as either modular or scalable. For the two platforms considered, the components common between the platforms happen to be more readily obtained off-the-shelf than fabricated. For this example, the shared components exactly coincide with the modular components and those unique to each platform coincide with the scalable components.

A component library is now built to document the components identified. The relative simplicity of the platforms and small number of components to track permits the use of the Excel® spreadsheet application for this task. Table 1 shows two sample component library entries in order to illustrate the type of information being tracked. At this stage, the attributes of interest are simply stated along with their units of measure if applicable. As the product family becomes more developed, the library will be populated with alternatives of each component. Component alternatives are distinguished from one another by differences in their attributes. For example, two instances of propellers may be populated into the library: one with a 10 in. (254 mm) diameter and the other with a 12 in. (305 mm) diameter.

TABLE 1. SAMPLE ENTRIES IN THE COMPONENT LIBRARY FOR A MODULAR AND A SCALABLE COMPONENT

Component: Motor	***Classification***: Modular
Attributes	***Interfaces***
1) Manufacturer	a) Speed controller power wires
2) Model	b) Propeller mount
3) Kv rating (RPM/Volt)	c) Arm mount
4) Weight (lbs.)	
5) Body diameter (in.)	
6) Body height (in.)	
7) Shaft diameter (in.)	
8) Shaft height (in.)	
9) Bolt pattern small diameter (in.)	
10) Bolt pattern large diameter (in.)	
11) Base pad diameter (in.)	
12) Bolt thread diameter (mm)	

Component: Multirotor arm	***Classification***: Scalable
Attributes	***Interfaces***
1) Length (in.)	a) Hub connection
2) Weight (lbs.)	b) Motor mount
3) Material volume (in.^{3})	
4) Base height (in.)	
5) Base width (in.)	
6) Motor mount bolt hole diameter (mm)	

The ranges of the mission requirements in Figure 8 permit a rough sizing exercise of both multirotor and fixed wing platforms. The result of this exercise is a first guess at the size of propellers, motors, and batteries the vehicles will have. This in turn gives an indication of the size of the speed controllers, central hub and arms for the multirotor, and wing, fuselage, and flight control surfaces for the fixed wing. Candidates for multirotor components identified by this analysis are organized into a matrix of alternatives shown in Figure 10. Speed controllers, GPS, and flight controllers have been omitted from Figure 10 because speed controllers match one-to-one to motors and the same GPS and flight controller are used across all variants.

The component information in Figure 10 allows the first configuration layout to be generated for each platform in the form of model skeletons. The model skeleton for the multirotor is implemented in CATIA® V6 and is shown in the left side of Figure 6. The multirotor skeleton is populated with locations of the interfaces between parts such as where the arm meets the hub, and also with environmental boundaries such as the plane where the ground exists when the vehicle is not airborne. Capturing as much information in the model skeleton as possible is beneficial as it will be leveraged by logical rules to enforce deign logic. For example, the ground plane is later used to check whether the vehicle will be stable and sit level when it is on the ground.

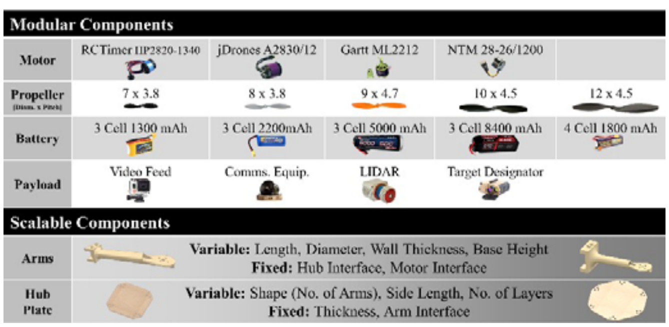
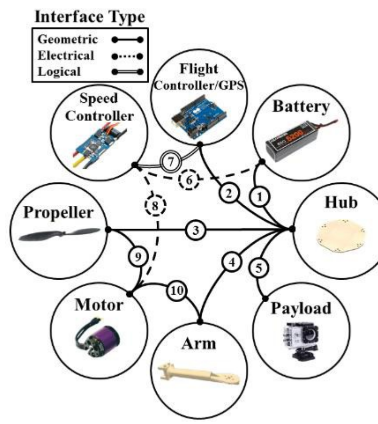


FIGURE 10. MULTIROTOR PLATFORM COMPONENT MATRIX OF ALTERNATIVES

Interface Design

The required components identified by the functional decomposition for the multirotor platform interface with one another in various ways. The network of interfaces between these components is depicted in Figure 11. Standards for each interface are developed as follows.

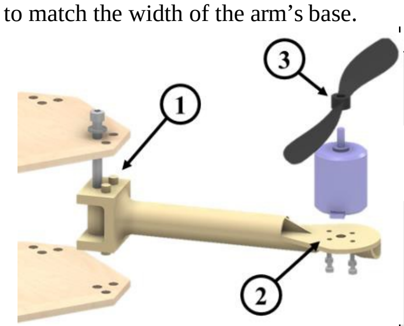
The modular components include the motor, propeller, battery, flight controller/GPS, servo motors, and speed controllers. By definition, these parts will not be modified during the design process. Parts interfacing with these components must conform to their pre-established interfaces. Several examples are the motor mount bolt geometry, the motor's electrical connectors, the motor shaft-propeller connection, and the propeller swept disc area. The motor mount bolt geometry consists of four M3 screws positioned around a central shaft hole. All motors share this pattern but the spacing between the holes varies motor to motor. The standard for the motor-arm interface geometry (Figure 11, interface 10) is therefore defined as the pattern of four bolt holes visible in Figure 12. The spacing between the bolt holes is left as a parameter so that the design can be updated when a different motor is selected. The motor's electrical connectors are of a specific type, size, and shape. This sets the standard for the motor-speed controller electrical interface (Figure 11, interface 8). The motor shaft is circular and has a specific diameter. Thus, the standard interface geometry between the propeller and motor (Figure 11, interface 9) is defined as a circle of that diameter. The propeller sweeps out a disc of diameter equal to the propeller's diameter. This forms an interface with the hub (Figure 11, interface 3) in that the propeller must be far enough away from the hub, with some margin, to prevent interference between the parts. Additionally, parts mounted on the hub cannot overhang its edge into this swept disc.



Interface	*Description*
1	Battery mount on hub
2	Flight controller mount on hub
3	Propeller swept disc interference
4	Hub-arm connection
5	Payload mount on hub
6	Propulsion power supply
7	Throttle signal
8	Motor drive power
9	Propeller mount on motor
10	Motor mount on arm

FIGURE 11. MULTIROTOR PLATFORM INTERFACES

Other interfaces have degrees of freedom left to the designer. Design automation will be faster and encounter fewer errors if the number of degrees of freedom is reduced. This is accomplished through defining custom interface standards. An example is the arm-hub interface geometry (Figure 11, interface 4). This interface design is geometric in nature and is left totally to the designer. A custom standard of two circular aligning tabs and a single through-hole for a bolt is defined and is visible in Figure 12. All variant designs conform to this standard, but the spacing of the tabs is left as a degree of freedom. The spacing will be automatically scaled for each new design to match the width of the arm's base.



Interface	*Description*
1	Arm-hub interface geometry
2	Motor-arm interface geometry
3	Motor-propeller interface diameter

FIGURE 12. EXPLODED VIEW SHOWING GEOMETRIC INTERFACES ON A MULTIROTOR PLATFORM

The fixed wing platform's fuselage, wing, and flight control subsystems are decomposed into several individual components. The configuration calls for parts assembled around a carbon fiber tube, which makes up the fuselage shaft. A battery cage, motor mount, component mounting plates, wing, and empennage mount slide onto the shaft and lock onto one another via alternating teeth. The geometric interfaces structure is shown in Figure 13. The electrical and logical interfaces are the same as the multirotor's, but with the addition of servo motors to drive the flight control surfaces.

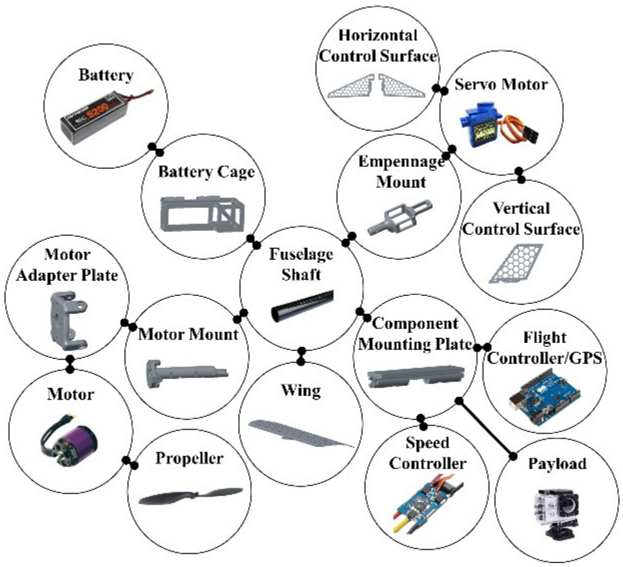


FIGURE 13: FIXED WING PLATFORM GEOMETRIC INTERFACES

The fixed wing motor mount, shown in Figure 14, is an example of a modular component-derived interface standard. It consists of two parts. A motor-specific adapter plate mates to a mount attached to the fuselage shaft via four additional screws. This multi-part assembly converts interface geometry of any motor to a common geometry useable in all variant designs.

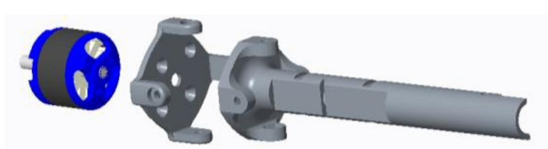


FIGURE 14: EXPLODED VIEW OF FIXED WING MOTOR MOUNT ASSEMBLY

Concept Refinement and Design

As a stakeholder, the REF presents a manufacturing constraint which limits part sizes. The print bed tray size of the 3-D printer to be used is limited to 8 in. by 6 in. by 6 in. (203 mm by 152 mm by 152 mm) width x length x height. Another consideration associated with the 3-D printer is the print direction. Bending strength is degraded for bending displacements in and out of the width-length print plane. Thus, a part printed with a load bearing feature primarily in the vertical 6 in. direction has degraded structural integrity. The combination of these two 3-D printing considerations results in a constraint stating the multirotor arm length be no longer than 8 in.

Figure 15 depicts the executable model-based techniques developed to automate multirotor SUAS design. Arrows indicate the linking structure between elements, with arrows in the upper right indicating information feed-forwards and arrows in the lower left indicating information feedbacks. Elements in Figure 15 preceding the physical model constitute the conceptual and preliminary modeling tools while the physical model embodies the detailed design and modeling tools.

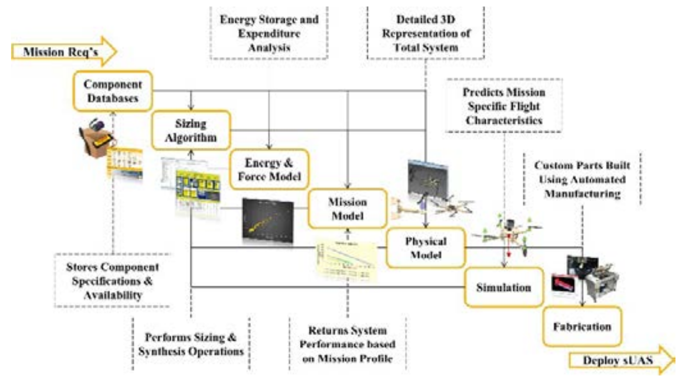


FIGURE 15. MODEL-BASED DESIGN PROCESS ARCHITECTURE FOR THE MULTIROTOR SUAS PLATFORM

Commercial CAD packages surveyed for used in this research effort are able to interface directly with Excel® spreadsheets. Therefore, both the component library and sizing algorithm are developed in Excel® and coded in Visual Basic®. The sizing algorithm takes information from the component library to perform a full factorial search over all combinations of component alternatives. Optimal design variables corresponding to each combination are developed, and then those designs are evaluated in a force and energy based model derived using the mission's flight profile. In order to reduce model error, the thrust and power consumption estimations are interpolated from test data gathered by the team using the actual components in the library. Those combinations of components that do not meet requirements are filtered out and the remaining combinations are ranked using the Technique for Ordered Preference by Similarity to Ideal Solution (TOPSIS), a well-known multi-attribute decision making technique [23].

The highest ranked combination becomes the design variant by default. However, the user is able to select a different combination if desired. The variant's component combination and its design parameters are then passed to the detailed design tools. The tools are implemented in CATIA® for 3-D modeling and CATIA's KnowledgeWare® toolset to encode logical rules. The logical rules first search a repository of 3-D models to find and insert the selected instances of modular components into the main 3-D model. Then, driving design parameters such as arm length, hub width, and number of hub layers are pushed to the model skeleton, which is updated accordingly. At this point, logical rules parse the model, performing operations such as enforcing design logic that eliminates invalid geometry, scaling interface geometry to match the modular components, repositioning parts to fit on the hub, modifying structures to improve strength and save weight, and filleting 3-D printed parts as required for manufacturing. The multirotor arms and the fixed wing's wing sections, control surfaces, and component mounts/adapter plates are designated as 3-D printed components.

Figure 16 illustrates the multirotor model before (top) and after (bottom) a design update. The top multirotor is the model in a default state, designed for generic requirements. Inputting new requirements into the design tools immediately triggers the tools to find different

modular components, resize the design parameters, and update the 3-D model resulting in the design shown on the bottom of Figure 16.

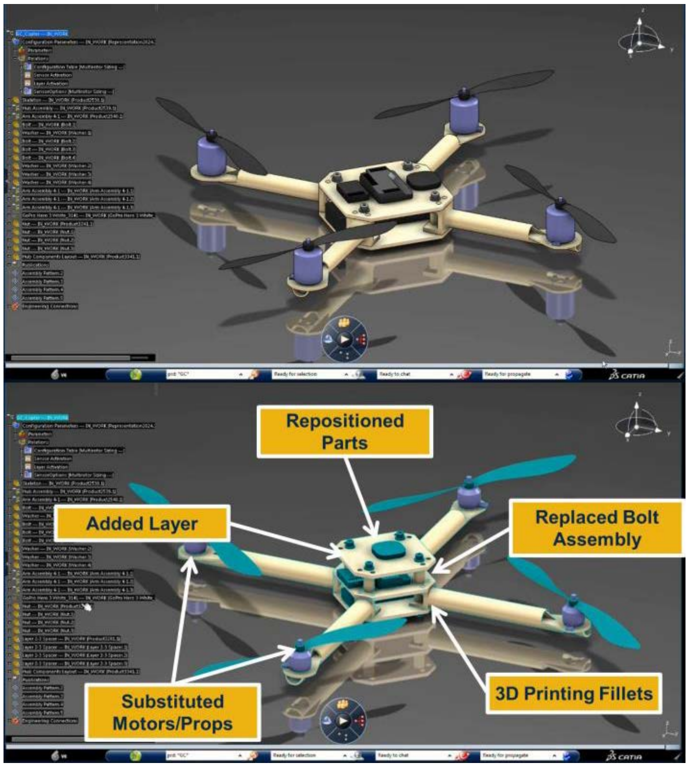


FIGURE 16. DEPICTION OF AN AUTOMATED MULTIROTOR MODEL UPDATE AS A RESULT OF A CHANGE IN REQUIREMENTS

Figure 17 shows a similar update for the fixed wing SUAS. Control surface span and chord, wing span, length, and airfoil, and battery cage dimensions are scalable in this platform.

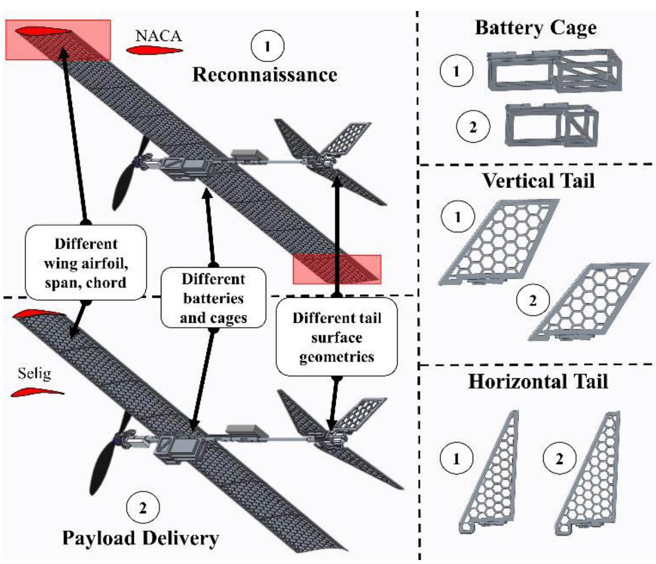


FIGURE 17: DEPICTION OF A FIXED WING MODEL UPDATE AS A RESULT OF A CHANGE IN REQUIREMENTS

Method Validation

Validation of the method was achieved by inputting mission requirements into the design tools and manufacturing and testing the resulting design. The final products were inspected and flight tested so that their characteristics and performance could be compared to the input requirements. Two multirotor designs were produced: one for a short range reconnaissance mission and another for a payload ferry mission. The requirements for each mission are given in the "Target Value" columns of Table 2 and Table 3 respectively. The "Returned Design" columns are the values predicted by the design tools for the resulting design. Two fixed wing SUAS were also produced for similar missions using requirements specific to the fixed wing platform.

TABLE 2. SHORT RANGE RECONNAISSANCE MISSION REQUIREMENTS AND RESULTING DESIGN VALUES

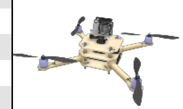
Requirement	*Target Value*	*Returned Design*	*Returned 3-D Model*
Max. Outer Dimension (in.)	33.0	29.7	
Max. Weight (lbf.)	5.0	3.08	
Min. Endurance (minutes)	10.0	12.1	
Max. Build Time (hrs.)	22.0	18.0	
Payload Capacity (lbf.)	0.0	0.99	
Sensor	GoPro	GoPro®	

TABLE 3. PAYLOAD FERRY MISSION REQUIREMENTS AND RESULTING DESIGN VALUES

Requirement	*Target Value*	*Returned Design*	*Returned 3-D Model*
Max. Outer Dimension (in.)	50.0	34.1	
Max. Weight (lbf.)	20.0	5.16	
Min. Endurance (minutes)	7.0	7.18	
Max. Build Time (hrs.)	40.0	33.1	
Payload Capacity (lbf.)	4.0	8.63	
Sensor	none	none	

Table 4 compares the predicted values for each metric of the short range reconnaissance mission design (reproduced from Table 2) to the values obtained by building and testing the design. Figure 18 shows a flight test of this design with a still frame taken from the GoPro® camera feed. Figure 19 shows a fixed wing SUAS built for a similar reconnaissance mission. The results in Table 4 show that the ADAPt Design approach produced a design that met all the geometric and performance requirements. The performance of the as-built SUAS was conservative in that both weight and endurance exceed the requirements and the SUAS was able to perform aggressive flight maneuvers while carrying its camera payload. This is by design – models were built with conservative margins to avoid producing a SUAS that failed to meet mission requirements. However, the scale of the deviations between predicted and actual performance highlight a limitation of ADAPt Design, which is that it relies on very high modeling accuracy. The endurance model used is derived from a first-principles energy balance and a simple hover-only mission model. Build time is underestimated due to underestimating the time required to dissolve the specific type of 3-D printed support material used. In both cases, what may seem like insignificant assumptions or inaccuracies in modeling result in large discrepancies between predicted and actual performance of the as-built vehicle.

TABLE 4. COMPARISON BETWEEN REQUIRED, PREDICTED, AND MEASURED REQUIREMENTS FOR THE SHORT RANGE RECONNAISSANCE MULTIROTOR SUAS

Requirement Metric	*Required*	*Predicted*	*Actual*	*Error (predicted vs. actual)*
Outer Dimension (in.)	≤ 33.0	29.7	29.7	0.0%
Weight (lbf.)	≤ 5.0	3.08	2.99	3.0%
Endurance (minutes)	≥ 10.0	12.1	15.1	19.9%
Build Time (hrs.)	≤ 22.0	18.0	20.5	12.2%

FIGURE 18. SHORT RANGE RECONNAISSANCE MULTIROTOR SUAS PERFORMING IN-FLIGHT MANEUVER WITH CAMERA FEED

FIGURE 19: ASSEMBLED FIXED WING SUAS

CONCLUSION

Modern military operations expose Soldiers to rapidly evolving and often unforeseen problems. The nature of these operations suggests that SUAS can provide crucial intelligence to Soldiers in a timely manner. However, equipping Soldiers with SUAS assets to meet unforeseen needs poses design and logistical challenges. A solution is to design and develop custom-tailored SUAS at the site of deployment. This on-demand approach is enabled by an automated SUAS design capability.

The focus of this work is to develop a framework to define and architect a set of product platforms that are highly compatible with design automation. The framework developed, called ADAPt Design, leverages concepts from product family design, reconfigurable system design, and systems engineering to enable on-demand design and production of SUAS. Applications to both multirotor and fixed wing SUAS prove the method's ability to generate differing designs given contrasting requirements, and that designs meet their respective requirements. However, flight tests indicate that the design approach is in part limited by the accuracy of the underlying models. Future efforts will focus on improving the accuracy of the multirotor mission model to reduce the error margins observed in initial tests. Additionally, flight tests of the fixed wing SUAS designs will be used to validate the method's ability to generate feasible designs of dissimilar platforms.

Even though ADAPt Design was developed around small systems, the method could be applied to architect adaptable subsystem designs within more complicated products. The authors believe that the method is relatively scalable, and that it could be modified to account for increased product complexity.

REFERENCES

[1] Johnson, D. E., Markel, M.W., and Shannon, B., 2013, "The 2008 Battle of Sadr City: Reimagining Urban Combat," RAND Corporation, Santa Monica, CA.

[2] 2010, "U.S. Army Roadmap for Unmanned Aerial Systems 2010-2035," U. S. Army UAS Center of Excellence, Fort Rucker, AL.

[3] US Army Training and Doctrine Command (TRADOC), "The U.S. Army Operating Concept; Win in a Complex World," Ft. Eustis, VA, TP 525-3-1, 2014.

[4] 2006, "Army Unmanned Aircraft System Operations," Field Manual Interim No. 3-04.155, Headquarters, Department of the Army, Washington, DC.

[5] Piper, Staff Sgt. R., 2005, "Small UAV Provides Eyes in the Sky for Battalions," Army News Service.

[6] "Rapid Equipping Force," US Military, [Online]. Available: http://www.ref.army.mil/exlab.html. [Accessed 21 January 2015].

[7] Mian, Z. T., 2011, "A Multidisciplinary Framework for Mission Effectiveness Quantification and Assessment of Micro Autonomous Systems and Technologies," Dissertation, Georgia Institute of Technology, Atlanta, GA.

[8] Mian, Z. T., Dees, P., Hall, L., and Mavris, D., 2013, "Development and Implementation of Micro Autonomous Systems and Technologies (MAST) Interactive Reconfigurable Matrix of Alternatives (M-IRMA) for Concept Selection," Procedia Computer Science, Vol. 16, 213, pp. 708–717.

[9] Mangum, P. Jr., Fisher, Z., Cooksey, K. D., Mavris, D., Spero, E., Gerdes, J. W., 2015, "An Automated Approach to the Design of Small Aerial Systems Using Rapid Manufacturing," International Design Engineering Technical Conferences, Vol. 2B, Paper No. DETC2015-47786.

[10] Jiao, J., Simpson, T., and Siddique, Z., 2007, "Product Family Design and Platform Based Product Development: a State-of-the-Art Review," Journal of Intelligent Manufacturing, vol. 18, pp. 5-29.

[11] Simpson, T.W.,Maier, J.R.A., and Mistree, F, 2001, "Product Platform Design: Method and Application," Research in Engineering Design, Vol. 13, 1, pp. 2–22.

[12] Simpson, T. W., Jiao, J., Siddique, Z., and Hölttä-Otto, K., 2014, Advances in Product Family and Product Platform Design, Springer Science and Business Media, New York, Chap. 1.

[13] Li, L., Huang, G. Q., and Newman, S. T., 2007, "Interweaving Genetic Programming and Genetic Algorithm for Structural and Parametric Optimization in Adaptive Platform Product Customization," Robotics and Computer-Integrated Manufacturing, Vol. 23, 6, pp. 650–658.

[14] Ulrich, K., and Tung, K., 1991, “Fundaments of Product Modularity,” Issues in Mechanical Design International, pp. 73–79.
[15] Ulrich, K., and Eppinger, S. D., 1995, “Product Design and Development,” McGraw-Hill, New York.
[16] Arena, J., Allison, J., 2014, “Solving the Reconfigurable Design Problem for Multiability With Application to Robotic Systems,” International Design Engineering Technical Conferences, Vol. 2B, Paper No. DETC2014-35314.
[17] Olewnik, A., Brauen, T., Ferguson, S., and Lewis, K., 2004, “A Framework for Flexible Systems and its Implementation in Multiattribute Decision Making”. Journal of Mechanical Design, Transactions of the ASME, 126(3), pp. 412–419.
[18] Literman, B., Cormier, P., Lewis, K., 2012, “Concept Analysis for Reconfigurable Products,” International Design Engineering Technical Conferences, Vol. 3, Paper No. DETC2012-71029.
[19] Patterson, M., Pate, D., German, B., 2011, “Performance Flexibility of a Reconfigurable Family of UAVs”. AIAA Aviation Technology, Integration, and Operations Conference, Paper No. AIAA 2011-6851.
[20] Forsberg, K. and Mooz, H, 1998, “System Engineering for Faster, Cheaper, Better,” Center for Systems Management.
[21] Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. A. M. T., 2002, “A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II.” Evolutionary Computation, IEEE Transactions on, Vol. 6(2), pp. 182-197.
[22] Cheng, A., Fisher, Z., Gautier, R., Cooksey, K. D., Beals, N., Mavris, D., 2016, “A Model-Based Approach to the Automated Design of Micro-Autonomous Multirotor Vehicle Systems,” American Helicopter Society 72nd Annual Forum and Technology Display.
[23] C.L. Hwang and K. Yoon, 1981, *Multiple Attributes Decision Making Methods and Applications*, Springer, Berlin Heidelberg.

A Model-Based Approach to the Automated Design of Micro-Autonomous Multirotor Vehicle Systems

Arthur Cheng
Graduate Research Assistant

Zachary Fisher
Graduate Research Assistant

Raphaël Gautier
Graduate Research Assistant

K. Daniel Cooksey*
Research Engineer

Dimitri Mavris
Regents Professor
Georgia Institute of Technology
Atlanta, Georgia, United States

Nathan Beals
U.S. Army Research Laboratory
Aberdeen, Maryland, United States

ABSTRACT

Small unmanned aircraft systems provide a novel means to improve situational awareness via surveillance and reconnaissance for military operations at the squad level. In order to provide mission-capable assets to soldiers quickly and reduce logistical burden, an automated, model-based approach is presented for the design and manufacture of small unmanned aircraft systems. This design methodology uses performance and geometry requirements provided by the end-user as constraints on a full-factorial set of potential vehicle alternatives constructed from an inventory of modular and scalable vehicle components. The chosen feasible vehicle alternative is then automatically modeled in a computer aided engineering/design tool and manufactured using additive or other rapid manufacturing techniques. An asset is delivered back to the user within hours of the initial request. This paper describes the methodology in detail including the role of interfaces, logical rules embedded in the process, and error propagation in the modeling environment. Finally, it presents the results of flight tests of an output vehicle in order to validate the integrated modeling and manufacturing method. Vehicle endurance measured during the flight tests were in reasonable agreement with performance predictions provided by analytical and empirical models during the design process. By improving these models, a process which guarantees a mission-capable vehicle can be realized.

INTRODUCTION

The U.S. Army recognizes that the modern battlespace is a rapidly evolving environment. Rapid and effective responsiveness from Soldiers and their equipment is required to maintain dominance. In light of this, improving the ability of Soldiers to respond to rapidly changing situations is a focal point of the "U.S. Army Operating Concept" from the Training and Doctrine Command (TRADOC) (Ref. 1). This point is corroborated by an assessment of U.S. military operations in the suburbs of Baghdad, Iraq conducted by the RAND Corporation for the U.S. Army. The RAND Corporation concluded that modern combat operations increasingly require decentralized decision making. It states,

> "The enemy is fleeting, which means that decentralized decision making is required. Units at the brigade level and below must therefore have access to the information and other capabilities required to support the rapid decisions necessary to deal with a highly mobile enemy … and to enable effective, independent action (Ref. 2)"

Both the TRADOC and the U.S. Army unmanned aircraft systems roadmap for 2010-2035 support this conclusion (Ref. 1, 3). The roadmap further recommends that small unmanned aircraft systems (SUAS) be used to enable decentralized decision making. It states,

> "UAS require and enable accelerated multi-echelon, decentralized decision-making, and execution, significantly changing the tempo and dynamics of operations. Lower echelon leadership must be empowered with authority and bandwidth to employ UAS as their changing situation dictates, operating at a tempo that is faster than higher echelon leadership can affect (Ref. 3)"

Presented at the AHS 72nd Annual Forum, West Palm Beach, Florida, USA, May 17-19, 2016.

SUAS can provide a means to develop the improved situational understanding required to support decentralized decision making during future U.S. Army operations. Accordingly, SUAS have been increasingly used by the U.S. Army. SUAS can perform functions such as intelligence, surveillance, and reconnaissance (ISR), security, manned-unmanned teaming (MUM-T), communications relay (Ref. 4), finding IEDs, identifying enemy combatants (Ref. 5), and supporting movement of supplies. Most importantly, SUAS can perform these functions with greatly reduced risk to the Soldier (Ref. 4). SUAS that have been deployed in the U.S. Army including the AeroVironment RQ-11 Raven have already proven themselves effective in providing situational awareness (Ref. 4, 5). These recent successful applications of SUAS attest to their effectiveness and the Army has expressed interest in expanding their use in future operations.

Future operations would benefit from the widespread employment of SUAS assets by Soldiers in the field at the battalion level and below. These assets would be available to Soldiers on-demand to acquire Actionable Intelligence in real-time. However, new vehicle assets need to be developed in order to equip Soldiers with SUAS. This can be accomplished by one of three general approaches:

1) **Multi-mission asset**: One SUAS asset covers all mission needs at the cost of diminished performance across all missions

2) **Set of optimized assets**: A set of SUAS assets are designed for a subset of specific missions are deployed; troops carry a number of assets to cover a range of possible missions

3) **Asset on-demand**: One-off SUAS asset is specifically tailored to and custom manufactured for the mission it will perform

These approaches are depicted in Figure 1 which shows each approach's mission coverage via three notional performance metrics. The grey square represents a multi-mission asset designed to operate in the center of the space. This vehicle has been designed to a set of mission requirements determined in advance of the vehicle's operational deployment. Therefore, its best payload capacity, range, and external dimensions are fixed from the perspective of the Soldier who is using it. Furthermore, a compromise exists between these three performance metrics causing the SUAS to be relatively inefficient in off-design missions.

The black dots show a set of optimized assets occupying discrete points in a slightly wider space. Individual black dots represent assets that are more specialized and therefore more efficient in missions at the extremes of the mission space. However, this comes at the cost of the logistical burden of the battalion transporting and storing a number of SUAS assets during operations.

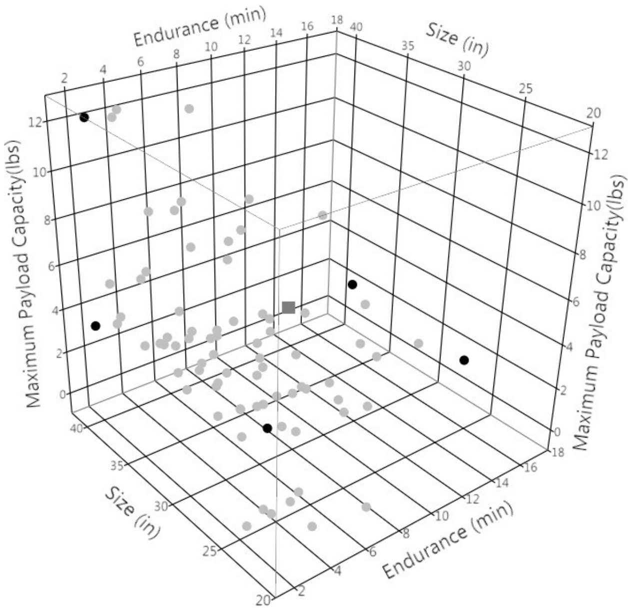


Figure 1. Mission capability coverage of three generalized approaches to SUAS development depicted using three notional performance metrics.

An on-demand design approach is readily explained via an analogy to Lego®. Figure 2 shows a box of Lego® bricks where modular parts can be constructed into different models depending on the desired product. Instructions are provided to help the user build different models out of the same set of components. In an on-demand approach, a small set of off-the-shelf parts which are difficult to manufacture on site (e.g. motors, propellers, and control electronics) will be provided ahead of time to a supply facility at a forward operating base. At the forward operating base is the U.S. Army Rapid Equipping Force (REF), a team of trained personnel equipped with mobile manufacturing equipment (Ref. 6). The REF has access to Expeditionary Labs (Ex Labs) which contain computer controlled manufacturing equipment including 3-D printers, consumer-focused computer numerical control (CNC) mills and laser cutters. Given a mission need, the REF combines off-the-shelf parts with parts manufactured on-site using automated engineering analysis and design tools. The product is a custom tailored SUAS to meet a specific need.

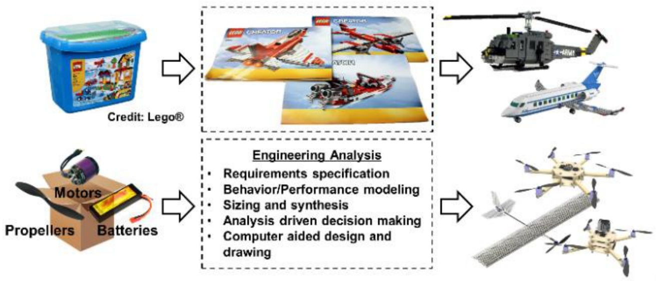


Figure 2. On-demand design philosophy illustrated via an analogy to Lego® (Ref. 7)

The gray dots in Figure 1 show designs that are generated on-demand and tailored to the need at hand. These assets exhibit performance that matches or exceeds a given mission need. Figure 1 illustrates that an on-demand approach captures the best of the multi-mission and optimized assets approaches. It covers a diverse range of mission needs, and assets are only generated when a new need is presented to mitigate the logistical burden of carrying multiple assets. This approach permits the Soldiers using the SUAS assets to decide which design best meets their needs, increasing their ability to respond to unforeseen situations.

Research Objectives

Several challenges must be addressed before an on-demand approach to SUAS development can be realized. The first challenge stems from the desire for rapid responsiveness. Designs generated on-demand need to be generated quickly, and so design activities including engineering analyses need to be automated and integrated into a streamlined workflow. Furthermore, the inputs to the workflow should be capabilities and performance requirements that map directly to a Soldier's operational needs.

Perhaps the most critical challenge is that each SUAS design will not be tested before operational use. The intent is to be able to move directly from design to manufacture and then to deployment, implying that the SUAS must perform as intended with limited or no system testing. This challenge is best illustrated through the systems engineering "vee" model in Figure 3. The "vee" model in the form presented by Forsberg and Mooz captures the systems engineering actions of a general development cycle (Ref. 8). In the context of SUAS development, the left leg involves translating capabilities and user requirements into system requirements, and then decomposing the design into increasingly specific subsystems. At each step, a plan is established to verify and validate the resulting subsystems and the total system itself. When it is time to follow the right side of the "vee", traditional design processes rely on having an assembled product to conduct verification and validation. This is not the case in an on-demand approach, where the re-composition leg of the "vee" will have to be collapsed into the decomposition leg. This is possible though pre-validation of platform architectures, configurations, components, and subsystems, as well as leveraging computer based modeling tools and automated manufacturing techniques. A key conclusion is that the design processes need to be carefully architected in order to achieve all of the verification and validation steps.

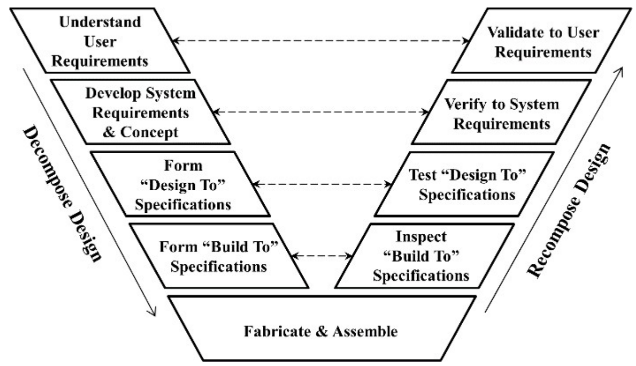


Figure 3. Forsberg and Mooz Systems Engineering "Vee" Model (Ref. 8).

In order to address these challenges, Georgia Institute of Technology and the U.S. Army Research Laboratory (ARL) have collaborated in the Micro Autonomous Systems Research (MASR) project. MASR is a multi-year effort to research capabilities for assessing the operational utility of small autonomous systems assisting at the squad level, and improving systems engineering processes for the development of these systems. The MASR effort has led to the development of systems engineering processes to enable an on-demand approach to SUAS design. Reference 9 details the capability to automatically generate a design for use within a building's interior.

Subsequent work has led to the development of a method for architecting product platforms that are highly compatible with design automation. The method has been termed "Aggregate Derivative Approach to Product Design" (ADAPt Design) reflecting the way new designs are generated: aggregations of modular components are integrated via custom manufactured scalable components to form a design variant. Reference 7 provides an introduction to the ADAPt Design method. This paper describes in detail the application of ADAPt Design to a multirotor SUAS including the SUAS platform design and the supporting automated design tools.

Product Family and Product Platform Design

ADAPt Design leverages concepts and terms from the field of product family design. Usage of these terms and concepts vary in the literature. This section establishes a consistent lexicon for the context of this paper.

A **product family** is a group of similar products derived from a common platform. Individual products belonging to a family are called **variants**, and each variant has a set of distinguishing features which allow it to meet different

requirements than other variants (Ref. 10, 11). The advantages of using product families to derive new designs lie in the reuse of major design elements, Design concepts, design logic, product functionality, physical layout, and many of the related design decisions are shared amongst all variants in a family. Reuse of these design elements reduces design time, improves design quality, increases product flexibility, and cuts program risk (Ref. 12).

Platforms are the groupings of "components, technologies, subsystems, processes, and interfaces" that form the basis from which variants in a family are derived (Ref. 1). Two types of platforms have been identified: **scalable platforms** where variants are produced by varying scalable design variables, and **modular platforms** where variants are produced through the exchange of different modules. A characteristic of modular platforms is a one-to-one mapping between functional element and physical components (Ref. 14).

An important semantic distinction is the one between product architecture and product configuration. **Product architecture** refers to the arrangement of functional elements into physical units and the interaction between these units (Ref. 15). **Product configuration** refers to the spatial layout of physical components, features, and modules. In the context of a product family, configuration defines the allocation of these elements between product variants (Ref. 16).

ADAPT DESIGN

Design automation of SUAS is enabled by extending the notion of a product family. The traditional concept of a product family is only concerned with a finite number of designs occupying discrete parts of the design space. The approach presented in this work treats the product variants not as discrete entities but as potential designs that vary continuously over a fixed design space. ADAPt Design is a method developed to implement this concept. Fundamentally, the method uses rigorous systems engineering techniques to form an executable link between input requirements and an output design. "Executable" is not intended to take on an abstract meaning in this context. Instead, it indicates that the link exists in executable code. The code takes requirements as inputs and uses them to directly drive computer aided engineering and design (CAE/CAD) tools which output detailed models and manufacturing files.

The outcomes of an application of ADAPt Design are: (1) the definition and architecture of a set of product platforms that are highly compatible with design automation and (2) automated design tools to drive CAE/CAD packages. The platforms are a hybrid modular and scalable architecture. Some components can be swapped one-for-one to form a new variant, while others have features that vary continuously. Figure 4 illustrates the ADAPt Design vision applied to a multirotor SUAS. A subset of modular parts is selected from a "shoebox" of alternatives, and then scalable integration parts are automatically designed. The combination of the modular and scalable parts form a design variant capable of meeting specified user requirements.

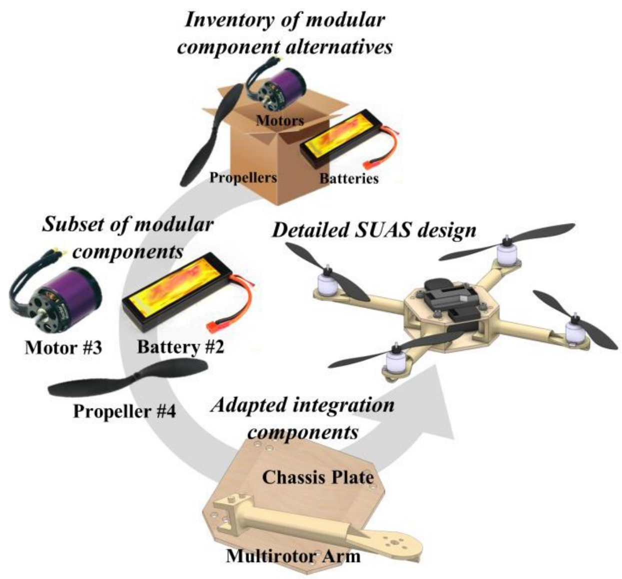


Figure 4. Illustration of ADAPt Design as applied to a multirotor SUAS platform

Like any design method, ADAPt Design is implemented iteratively and over time. For the purposes of presentation, the method is described in terms of four actions: (A) requirements analysis, (B) architecture selection, (C) interface design, and (D) concept refinement and design.

A. Requirements Analysis

The first action in ADAPt Design is to determine and document capabilities and associated system requirements to be addressed by the product family. The following items are identified:

1) Broad definitions of what objectives are to be addressed by the product family in terms of capabilities. Specifically, this means articulating clearly whose needs will be addressed along with a statement pertaining to how they will be addressed.

2) Key stakeholders in the product family's use and the requirements and constraints they impose on its development. These requirements and constraints will both bound and be used to validate the product family architecture.

3) Engineering metrics against which derived designs will be measured and compared. These are typically quantifiable characteristics of each design and may include performance metrics, physical dimensions, and required manufacturing details.

In the case of SUAS, the capability desired is to supply a Soldier with a tailored SUAS to an unforeseen need. In broad terms, an unforeseen need involves the SUAS flying a short mission while carrying a payload specified by the user. This statement is kept very general in order to avoid ruling out missions that are not typical for SUAS, however examples of mission types that are expected to be more common are:

- Exterior reconnaissance and surveillance
- Interior reconnaissance and surveillance
- Reconnaissance by fire
- Communications relay
- Logistics support and supplies ferry

Certain types of missions may require specialized sensor packages. Therefore, a set of standard sensors are identified as a camera, communications equipment, a LIDAR, and a target designator. Other than the Solider who requests the SUAS, the REF who will manufacture the SUAS is a stakeholder and imposes constraints on the product family. The designs produced must be compatible with the manufacturing processes available to the REF and must only use available components and materials.

Mission Requirements

A new design originates from high-level performance requirements and manufacturing constraints inputted by the user. High level requirements are used instead of detailed design requirements to simplify and accelerate the interactions between the user and the design tools.

For the multirotor SUAS, requirements and constraints include minimum endurance, maneuverability, maximum weight and size, and extra payload weight. Minimum endurance is a lower bound on the endurance of the vehicle calculated by finding the time the vehicle could operate at the minimum power required for flight. Minimum extra payload weight is the extra weight (in excess of empty weight) the vehicle will need to carry for the mission. The endurance requirement is evaluated assuming that the SUAS is carrying the minimum extra payload weight. If excess thrust is available to the SUAS, it may be able to lift more than the minimum extra payload weight at the cost of reduced endurance. Maneuverability is a qualitative scale with three categorical settings (Normal, High, or Acrobatic). Each category responds to a specific thrust margin between the total amount of available thrust and the amount of thrust needed by the vehicle to hover. Maximum weight and size are parameters referring to the overall SUAS weight and the largest external dimension of the vehicle. These requirements are evaluated using measures of effectiveness in the form of engineering metrics. Expected ranges of these metrics are given in Table 1.

Table 1. SUAS performance metrics and expected ranges of values.

Metric	Minimum	Maximum
Payload Capacity (lbf.)	0.1	12
Minimum Airspeed (MPH)	0	N/A
Maximum Airspeed (MPH)	N/A	60
Endurance (minutes)	5	60
Size (in.)	12	50

The user is also given the option to specify if designs with more than four arms should be considered. Additional arms add propulsion system redundancy and provide a means of improving maximum payload capacity. Figure 5 shows the user interface for the design tools. Mission requirements and manufacturing constraints are input into the fields when a new vehicle design is desired.

B. Architecture Selection

The architecture selection action involves identifying and

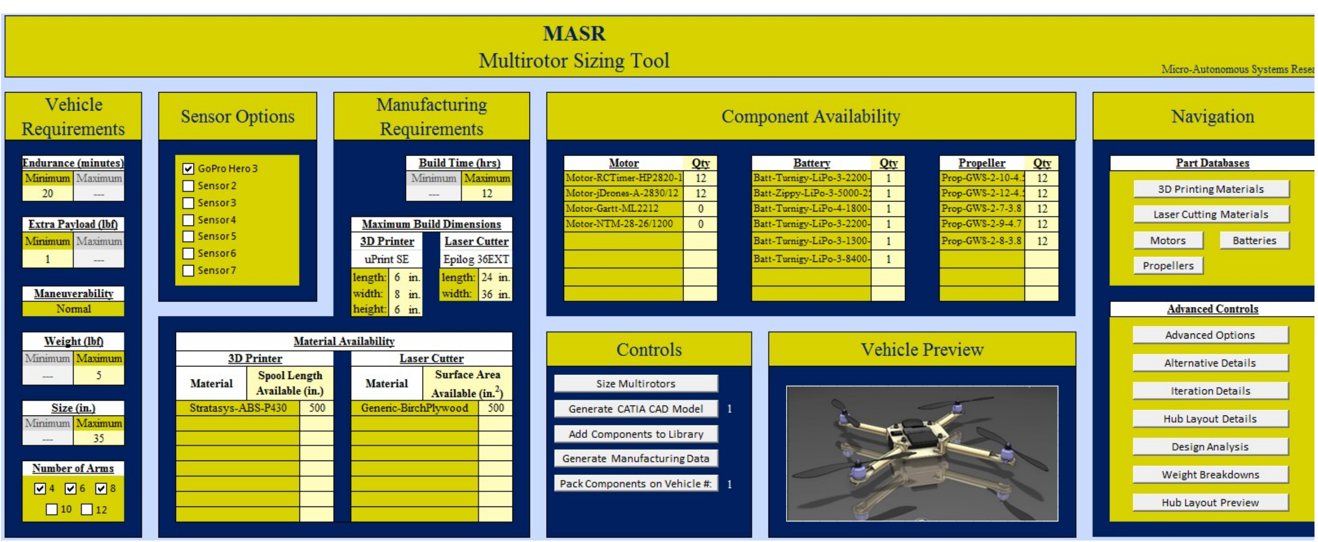


Figure 5. Multirotor design tool user interface.

defining the product platform(s) that will comprise the product family. This step begins with the objectives, capabilities, requirements, constraints, and metrics established during requirements analysis and concludes with high level configuration layouts of the platform(s).

SUAS Platforms

First, the platform(s) that will compose the product family are identified. In the context of SUAS, platforms correspond to vehicle types (e.g., rotary wing, fixed wing, flapping wing, etc). Later on when the product family design is made "executable", all variant designs are derived from one of the platforms defined in this step. It is therefore required that at least one platform be identified to cover each of the desired capabilities identified in requirements analysis.

Two platforms cover the desired capabilities: a hand launched fixed wing SUAS and multirotor SUAS. The fixed wing platform provides coverage of missions that require longer range and endurance while the multirotor platform covers missions with requirements for hover capability. This paper describes development of the multirotor platform. Further details about the fixed wing platform can be found in Ref. 17 .

Component Library

Next, a functional decomposition of the selected platforms is performed, and subsystems and individual components are allocated to each function. The resulting list of components and subsystems forms the platform architecture and a preliminary component library. A functional decomposition of a generic multirotor SUAS is shown in Figure 6.

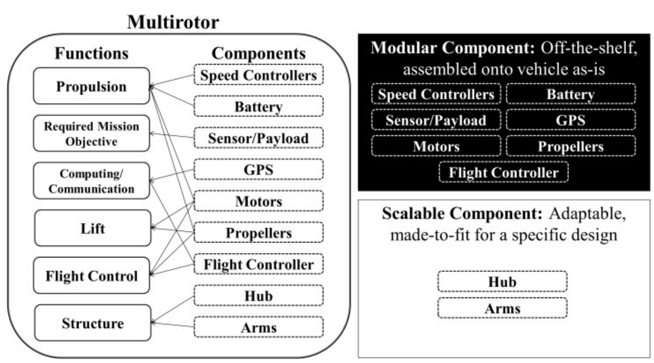


Figure 6. Multirotor SUAS functional decomposition and component allocation.

The purpose of the component library is to document the parts needed to build an instance of the platform, and to standardize the specification of individual components in a structured format. The component library serves as the primary source of information regarding the components available to form a design variant. This information is used by the automated model-based analyses. More precisely, constructing the component database consists of determining:

1) The attributes of each component. Attributes are the key characteristics of a component that are necessary to distinguish between component alternatives and model the component's performance. Scalable components have additional attributes that can be scaled within pre-defined ranges.

2) The classification of each component as either "modular" or "scalable." A modular component implies discrete alternatives of that type of component are swapped into new design variants with little or no modification. Scalable components are scaled via a limited set of continuous design parameters and are to be manufactured on-site. Consideration must be given to a tradeoff between design freedom and manufacturing complexity when classifying components. For example, defining a component as scalable results in increased flexibility of the designer to customize that part, a decreased amount of modular pieces to store, simplification of supply management, and decreased assembly complexity. However, having numerous scalable components may increase the manufacturing time. Furthermore, it may not be possible for certain components with complex geometry or specialized materials to be manufactured onsite.

3) The interfaces of each component. In the context of ADAPt Design, an interface is a specification of how two or more components interact with each other. For example, interfaces can be physical, electrical, or logical as in digital communication. After being textually described, interfaces are enforced within the executable design environment. Enforcing an interface may involve checking that two components are compatible (e.g., the physical dimensions of a modular/modular interface) or setting components' attributes in order to satisfy the interface (e.g., setting scalable parameters in a modular/scalable or scalable/scalable interface).

The combination of a subset of component alternatives results in a candidate design with certain mission capabilities. All possible combinations of component alternatives result in an overall mission capability coverage, which can be visualized by a cloud of points similar to the gray dots in Figure 1. The shape and density of this cloud result from the choice of component alternatives that populate the component database. Diverse component alternatives generate a sparse and expansive cloud that covers more missions. However, this comes at the cost of larger gaps between potential designs in the mission space and a reduced ability to exactly match a mission need. The inverse is true when component alternatives have a narrow range of specifications. A simultaneously expansive and dense cloud can be attained by incorporating many component alternatives, but this increases supply logistics complexity. Consideration must therefore be given to a trade-off between the number of alternatives of each component and the mission capability coverage.

The component library for the SUAS platform populated with alternatives for modular components is shown in Figure 7. Electronic speed controllers (ESC), flight controllers, and GPS units have been omitted because speed controllers map one-to-one with motors and only one alternative for flight controller/GPS unit are considered.

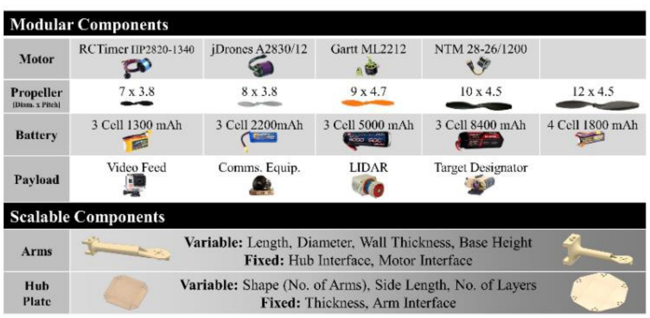
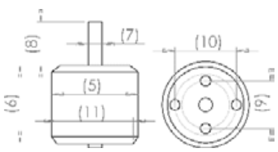


Figure 7. Multirotor SUAS component library (Ref. 7).

Table 2 shows the information tracked in the component library for the motor. Motors are classified as modular components to satisfy the requirements constraining manufacturing time to a few hours. Manufacturing a motor is a complex task requiring specialized parts, tools, and technical skills. Motors manufactured in-situ would be less reliable components than commercially available alternatives. The motor shares geometric interfaces with the arm and propeller. Satisfying the propeller interface consists of checking the compatibility of the motor's shaft height and diameter attributes with the propeller through hole diameter. Satisfying the arm interface consists of matching the size and placement of holes on the arm to the motor mount bolt pattern of the motor. The motor also shares an electrical interface with the speed controller. Satisfying this interface involves checking the motor's current draw with the speed controller's rated current.

Table 2. Component library entry for the motor.

Component: Motor	**Classification**: Modular
Attributes	**Interfaces**
1) Manufacturer	a) Speed controller current draw
2) Model	b) Propeller mount
3) Velocity constant (RPM/Volt)	c) Arm mount
4) Weight (lbs.)	
5) Body diameter (in.)	
6) Body height (in.)	
7) Shaft diameter (in.)	
8) Shaft height (in.)	
9) Bolt pattern small diameter (in.)	
10) Bolt pattern large diameter (in.)	
11) Base pad diameter (in.)	
12) Bolt thread diameter (mm)	

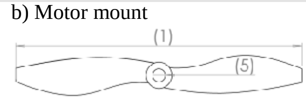


Propellers are also classified as modular components despite the fact that small-scale propellers can be manufactured using 3D-printers. The cost of performance and reliability compared to commercially available off-the-shelf propellers does not justify the use of 3-D printed propellers when a small number of propellers ranging from 7 in. to 12 in. in diameter are sufficient to cover a wide range of performance.

In addition to the motor/propeller interface, the propeller also interfaces with the chassis. The propeller swept disc must have no interference with the components placed on the vehicle's chassis. Although this interface links the propeller and the chassis, the length of the arm is the parameter that is changed in order to obtain clearance. The propeller's component library entry is given in Table 3.

Table 3. Component library entry for the propeller.

Component: Propeller	**Classification**: Modular
Attributes	**Interfaces**
1) Diameter (in.)	a) Chassis interference
2) Pitch (in.)	b) Motor mount
3) Number of blades	
4) Weight (lbs.)	
5) Through hole diameter (in.)	

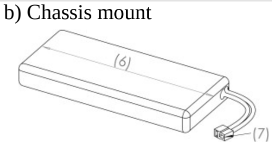


Electric components including the battery and ESC were chosen to be modular to improve reliability and decrease build time. Though electronic components may potentially be produced by the REF, this would require a large set of specific technical skills, as well as a much wider set of manufacturing machinery than what is available to the REF.

The battery has two interfaces: with the ESC and with the chassis. The ESC interface is electrical in nature. Satisfying this interface consists of checking that the battery's discharge current rating is below the ESC's maximum rated current and that the battery and the ESC use compatible electrical connectors. The interface with the chassis is geometric. This interface is satisfied if the chassis is large enough for the battery to fit on the chassis plate without overhanging the edge or interfering with other parts mounted on the chassis. Table 4 gives the battery's component library entry.

Table 4. Component library entry for the battery.

Component: Battery	**Classification**: Modular
Attributes	**Interfaces**
1) Cell chemistry (e.g. Lithium Polymer, Nickel Metal Hydride)	a) Speed controller current discharge
2) Capacity (mAh)	b) Chassis mount
3) Cell count	
4) Current discharge rating	
5) Weight (lbs.)	
6) External dimensions (in.)	
7) Connector type	

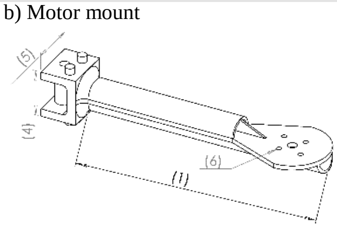


Arms are the structural components that link the chassis to the motor and are one-piece scalable components. The flexibility to modify the external dimensions and interface geometry of the arms is needed in order to accommodate different modular component alternatives. The arms are 3-D printed to achieve this flexibility. 3-D printing imposes a constraint as the print size of the 3-D printer is limited to 203 mm by 152 mm by 152 mm (width x length x height). This requires each arm of the quadcopter to be printed separately and independently from the chassis.

Arms interface with the motor and chassis. The motor pad on the arm must conform to the mount pattern of the selected motor alternative, and the interface geometry between the arm base and chassis must be consistent. The scalable nature of the arm allows flexibility in its length and structural geometry. In the case of missions where survivability may not be the prime requirement such as in a reconnaissance by fire mission, the material structure can be made lighter at the expense of durability. Table 5 gives the arm's component library entry.

Table 5. Component library entry for the arm.

Component: Arm	**Classification**: Scalable
Attributes	**Interfaces**
1) Length (in.)	a) Chassis mount
2) Weight (lbs.)	b) Motor mount
3) Material volume (in.3)	
4) Base height (in.)	
5) Base width (in.)	
6) Motor mount bolt hole diameter (mm)	



Chassis plates are scalable components. Geometric features of the chassis plates are primarily two-dimensional which permits the use of laser-cutting instead of 3-D printing as a rapid manufacturing technique. The result is a significant decrease in manufacturing time. Attributes include the plate length, width, thickness, and the total number of plates used. The chassis plates are allowed to scale in dimensions as well as number (as in adding layers) to accommodate the type of sensors and payloads required by the user.

The chassis plates have five geometric interfaces. Aside from the chassis/arm, chassis/propeller and chassis/battery interfaces that were previously discussed, the payload and flight controller are mounted on the chassis through two interfaces. Satisfying both of these interfaces consists of scaling the chassis plates' dimensions and numbers in order to fit all the components within the boundaries of the chassis.

<u>Configuration Layout</u>

Architecture selection concludes with the definition of preliminary configuration layouts, which capture rough spatial component placement and bounds on the interfaces between subsystems. This step abandons the traditional notion that "major design changes are frozen at the end of conceptual design." The intent of architecture selection is to identify platforms that are highly adaptable and so the conventional notion is counterproductive. Instead, the configuration layouts indicate which components and requirements will drive interfaces, the magnitude of variation expected for the interface parameters, and a rough idea of the location of components, subsystems, and interfaces.

Configuration layouts are most useful when documented in the form of "model skeletons," or 3-D representations of key geometric planes, points and shapes located in space. The model skeleton approach is known as a "top down design" by the CAD community. Figure 8 illustrates an example model skeleton and the multirotor SUAS it represents. The skeleton model at this point is preliminary. It will be revisited and revised throughout the remaining design activities.

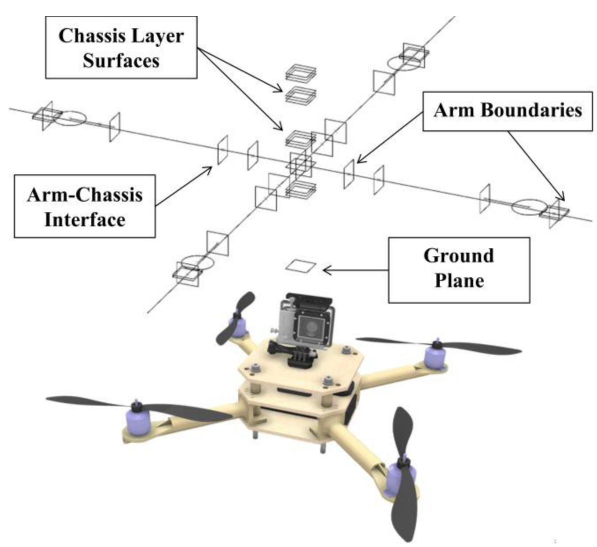
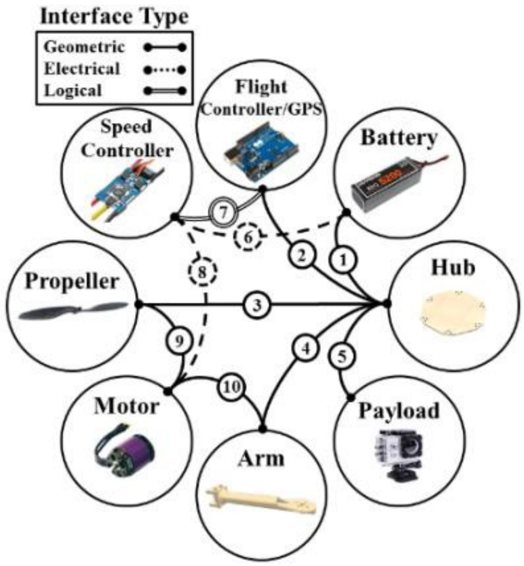


Figure 8. Model skeleton of a multirotor SUAS alongside the SUAS it represents.

C. INTERFACE DESIGN

Design automation is in part enabled by clearly describing interfaces between components. Interfaces can be geometric, electrical, or logical as in the case of digital communications. A characteristic of ADAPt Design which

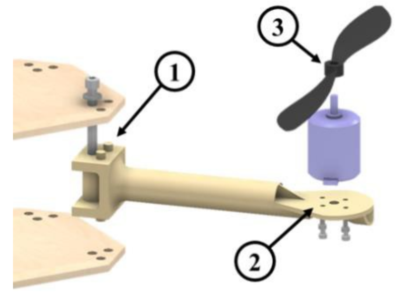
distinguishes it from traditional design processes is an early emphasis on interfaces. Well defined and consistent interface definitions provide a standardized mechanism to which new variant designs conform. At this stage, interface standards are set based on the selection of modular and scalable component types, and the configuration layouts defined previously. The major interfaces identified between the multirotor SUAS components are summarized in Figure 9.



	Interface	Type
1	Battery mount on chassis	Geometric
2	Flight controller mount on chassis	Geometric
3	Propeller swept disc interference	Geometric
4	Chassis-arm connection	Geometric
5	Payload mount on chassis	Geometric
6	Propulsion power supply	Electric
7	Throttle signal	Logical
8	Motor drive power	Electric
9	Propeller mount on motor	Geometric
10	Motor mount on arm	Geometric

Figure 9. Summary of major interfaces in the multirotor SUAS.

Modular components are by definition not modifiable. Therefore, the interfaces of these components are prescribed by those components. For example, the motor mount bolt pattern shown in Figure 10 item 1 is prescribed by the motors in the component library. These prescribed interfaces ae recorded as standards for the product family and are captured in both the component library and model skeletons. Control over other interfaces, including those between scalable parts, lies in the hands of the designer. An example is shown in Figure 10 as the arm-chassis interface geometry.



Interface	Description
1	Arm-chassis interface geometry
2	Motor-arm interface geometry
3	Motor-propeller interface diameter

Figure 10. Example interface definitions on a multirotor SUAS platform.

At this point, the designer leverages the configuration layouts to define custom interface standards. Often, interfaces between modular components and scalable components can be used to develop design logic to drive scalable parameters. For example, Figure 11 depicts the propeller radius driving the arm length. This is based on eliminating interference between the propeller's swept disc and the chassis. Similarly, the battery's largest dimension must be smaller than the width of the chassis. This creates a lower bound on the chassis' scalable width attribute.

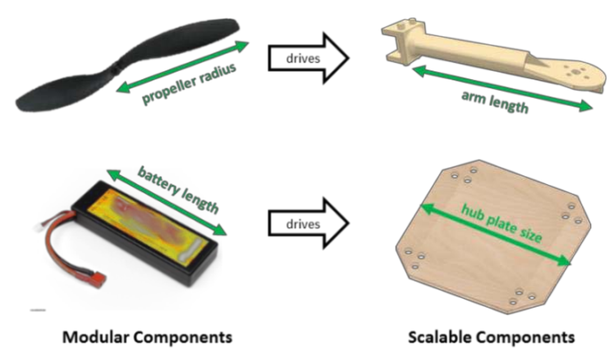


Figure 11. Modular components drive the design of scalable components.

At the conclusion of interface design, all interfaces between parts have a concrete definition. New geometric information such as locations of interfaces and parametric interface geometry are stored in the model skeleton.

D. Concept Refinement and Design

Concept refinement and design includes many of the design activities associated with traditional preliminary and detailed design, however in the context of ADAPt Design they differ in three ways. First, design activities are encoded into a chain of linked engineering analysis tools rather than being performed manually. The chain of tools takes customer needs in the form of capability requirements and automatically outputs a detailed set of models and

manufacturing files. Second, bills of materials and manufacturing techniques are specified pre-design rather than post-design. All logistics and manufacturing constraints are specified prior to design. Third, reduction in modeling error is critical. As described previously, the user expects to assemble and use the design immediately. Product verification and validation is therefore transferred from tests on the assembled SUAS to the models and analyses used in designing the SUAS. Accordingly, modeling error must be very low so that predictions closely match the actual behavior of the assembled product.

This step begins by refining the constraints identified during requirements analysis to define the complete set of constraints that the design tools will enforce. Constraints can be from different categories such as design, manufacturing, assembly, logistics, or regulatory, and should be quantified where possible.

For the multirotor SUAS, constraints include build time, maximum build dimensions, material availability, and component availability. Build time refers to the maximum amount of time needed to fabricate the vehicle including 3-D printing parts and assembling the vehicle. Maximum build dimensions define the largest dimensions the 3-D printer and the laser cutter can manufacture. Values for these constraints are specific to the model of the manufacturing machinery used. Material availability for the 3-D printer and the laser cutter describes the amount of filament available to be used in 3-D printing parts and the surface area of material available to be laser cut. Component availability specifies which motor, battery, and propeller alternatives are on hand to be used in a design.

Next, executable model-based design tools are developed. Multirotor SUAS design is readily divided into two types of models: preliminary and detailed design models. Preliminary modeling tools are responsible for determining which modular components alternatives are selected from the component library for use in the variant design. Candidate designs are synthesized by pairing combinations of modular component alternatives with values of high level design variables corresponding to scalable attributes. Detailed design modeling tools are responsible for translating design parameters and the selected component alternatives into manufacturable 3-D representations of parts. The detailed design tools are more than just 3-D models. They are paired with logical rules to enforce design logic and prevent impossible or invalid geometries. These rules can be conditional statements, checks, or iterative loops.

Figure 12 depicts a design structure matrix which summarizes the multirotor SUAS modeling tools and the links between them. Each box represents an element of the design tools and the arrows represent the flow of data between elements. Combined, the elements in Figure 12 form an automated, executable design cycle. The following sections describe how the major elements work and the flow of data between them.

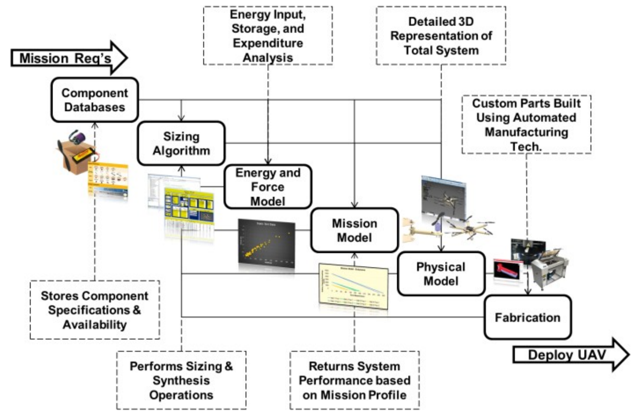


Figure 12. Design structure matrix for the automated multirotor design cycle.

<u>Sizing Algorithm</u>

A sizing algorithm serves as the central mechanism for automating conceptual and preliminary design activities. It coordinates the design cycle by calling upon other modeling elements. Given a set of performance requirements, the sizing algorithm uses information about available components from the component library and yields a ranked list of SUAS candidate designs. Each candidate's description can be passed to the detailed design tools to produce a physical model. The Sizing algorithm works by performing the following tasks:

1) Capture the user performance requirements and translate them into metrics usable within an automated process
2) Generate feasible designs through a full factorial search among modular component alternatives
3) Set the values of scalable components and ensure that interfaces and manufacturing constraints are satisfied
4) Filter out designs that do not fulfill user requirements and rank the remaining feasible designs based on user preferences

A more detailed depiction of the sizing algorithm's structure is shown in Figure 13. Each of the "Estimate" blocks in Figure 13 calls upon another modeling element from Figure 12. The order in which these modeling elements are called is the reverse order of the elements' computational complexity. This helps to reduce computation time so that the algorithm is capable of handling large full factorial search spaces.

The final step in the sizing algorithm is to score and rank the feasible alternatives. This is achieved using the Technique for Ordered Preference by Similarity to Ideal Solution (TOPSIS) (Ref. 18). TOPSIS is a well-known multi-attribute decision making (MADM) technique that was selected for its simplicity. A drawback of TOPSIS is that it requires user preferences that are often qualitative and rely

on subject matter expert input. Furthermore the rankings TOPSIS produces are highly sensitive to these preferences. Therefore, the design tools are programmed to return the top ranked candidate by default, but the user is given an override privilege in the event that a lower ranked design is desired.

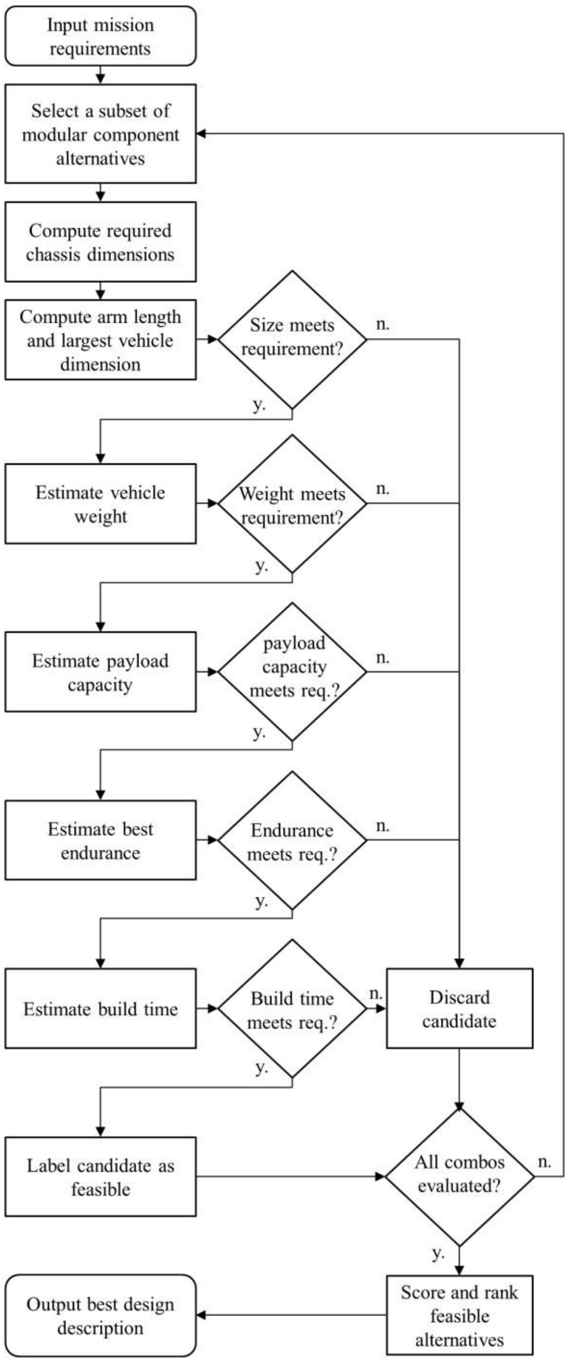


Figure 13. Sequence of the design logic in the sizing algorithm.

Energy & Force Modeling

Thrust and power consumption are the driving characteristics of vehicle performance and endurance, and are estimated using energy and force models. Two approaches to modeling thrust and power consumption were considered: first-principles physics-based models and data-driven tabular models.

In a first-principles physics-based approach, the basic physics behind each component is modeled to predict its performance. For example, thrust generated by a rotating propeller is modeled using a blade-element/momentum theory. This approach has the advantage of being applicable to any modular component alternative, even if it was not previously considered as an alternative in the component library. For example, a propeller with a different blade diameter and pitch can be represented using the same model simply by changing the blade geometry parameters in the calculations. Therefore, a first-principles physics-based model can be used to predict a wide variety of vehicle configurations with different components. However, as the number of interconnected components increases, accurately modeling the overall system becomes difficult. In addition, first-principles physics-based aerodynamics models do not scale well to the small scale of SUAS. Furthermore, hobby grade parts exhibit relatively high variability in performance between otherwise identical parts as a result of manufacturing deviations. These variations are difficult to predict using first-principles physics-based models.

A data-driven tabular modeling approach uses test data collected from experiments to map performance characteristics to combinations of components via lookup tables and interpolation. The advantages of this approach are that the complex interactions between components are captured in the recorded data. Additionally, variations due to manufacturing error are captured because the parts tested in the experiments are the exact parts that will be installed on the SUAS. This approach is simple to set up and manage provided the number of component alternatives is relatively small (on the order of three or four each of motors, propellers, and batteries). Disadvantages of this approach are that additional testing is required to incorporate new component alternatives into the model. Therefore, the model cannot be used to model component combinations that have not yet been tested.

A data-driven tabular model is implemented to achieve the high level of accuracy required to ensure modeled performance closely matches actual performance. The relatively small number of components in the library allows experimental data to be obtained for all possible combinations of the component alternatives. In the final implementation, the energy and force model consists of a data-driven tabular model in which discrete propeller-motor-battery combinations were tested and thrust curves were recorded. For each combination of propulsion system components, thrust values could be correlated to battery voltage, current draw, and motor RPM.

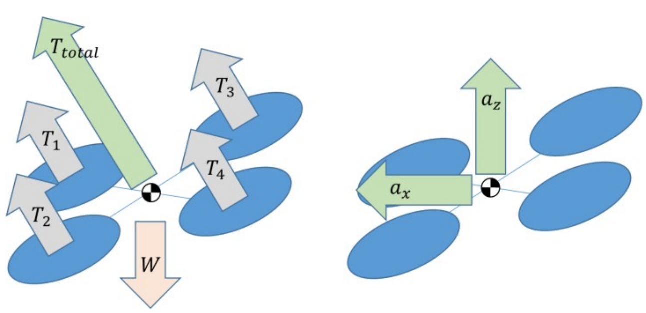
During the design process for the multirotor, a weight estimate of the vehicle in conjunction with a

maneuverability requirement is used to determine the thrust required of the vehicle. The data-driven tabular model is then queried to provide information on the propulsion system power consumption at the required thrust to determine vehicle endurance.

Mission Performance and Maneuverability Model

A mission performance and maneuverability model are used to estimate maneuverability and endurance characteristics of a candidate design in order to compare against performance requirements. The endurance model uses results from the energy and force models in conjunction with a mission profile to estimate the flight time of the quadcopter. As a first-order estimate, the assumption that a hover or slow-forward flight condition persists for most of the vehicle mission. This correlates to a constant power draw which is used to determine the vehicle endurance. It should be noted that the mission profile can be modified to contain a schedule of vehicle flight maneuvers over the mission duration such as climb, hover, cruise, and descent. Implementation of such a mission profile requires high-fidelity modeling of vehicle drag and was therefore not implemented in the prototype design tools.

The maneuverability requirement is quantified using instantaneous vertical and lateral acceleration of a hovering vehicle. The free body diagram shown in Figure 14 is used to establish a mapping between simultaneous vertical and horizontal acceleration capabilities and a required vehicle thrust to weight ratio. The required thrust to weight ratio is compared to a candidate design's maximum thrust to weight ratio using the tabular energy and force model.



(a) (b)

Figure 14. (a) Free body diagram and (b) kinetic diagram for a quadcopter executing an idealized instantaneous acceleration.

The analysis in Figure 14 assumes that the vehicle starts from a steady hover and instantaneously accelerates both vertically at an acceleration a_z and laterally at an acceleration a_x. In this scenario, the required thrust to weight ratio is given by

$$\frac{T}{W_{required}} = \sqrt{\left(\frac{a_x}{g}\right)^2 + \left(\frac{a_z}{g} + 1\right)^2}$$

In the above equation, g is the acceleration due to gravity. Figure 15 shows a plot of the required thrust to weight ratio over ranges of expected acceleration values.

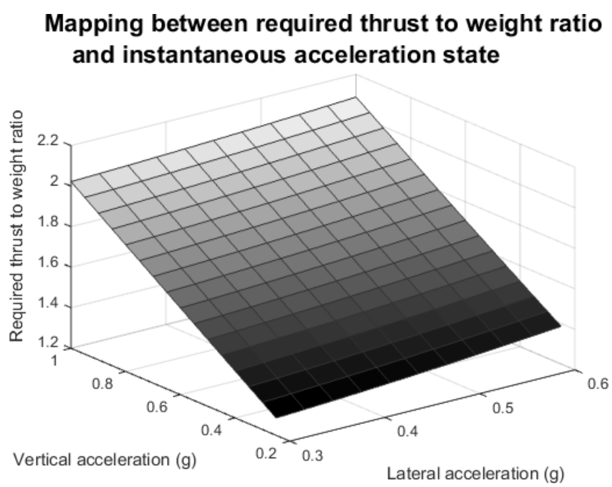


Figure 15. Plot of required thrust to weight ratio for a given simultaneous vertical and lateral acceleration.

The user is given three qualitative choices for the maneuverability requirement: "Normal", "High", and "Acrobatic". Classification of the maneuverability settings in terms of thrust to weight ratio are shown in Table 6.

Table 6. Vehicle maneuverability level classification based on the vehicle's maximum thrust to weight ratio.

Maneuverability setting	Maximum T/W
Normal	$T/W \geq 1.29$
High	$T/W \geq 1.66$
Acrobatic	$T/W \geq 2.09$

Physical Model

The physical model translates design parameters and the chosen component alternatives into a 3-D representation of the SUAS. The ultimate output of the physical model is a set of manufacturing files containing enough detail that the SUAS can be directly manufactured using rapid manufacturing techniques. The model is fully parametric; nearly all geometric features are derived by the model based on a combination of variable design parameters, equations, geometric relations, and design logic encoded into rules and checks.

Conceptually, the physical model can be decomposed into three major divisions: a model skeleton, logical rules and checks, and detailed geometry. The flow of information between these different parts is depicted in Figure 16.

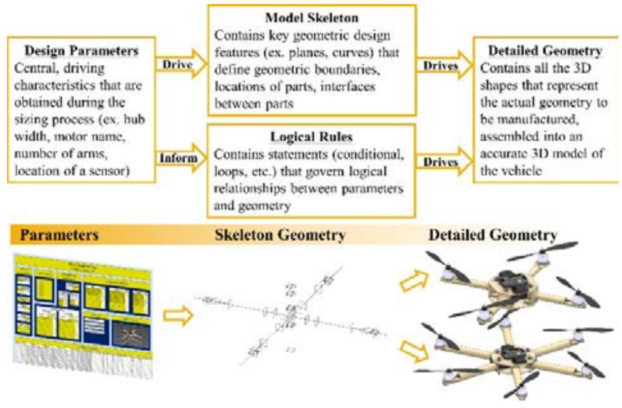


Figure 16. Major divisions of the physical model and the flow of design information between them (Ref. 7).

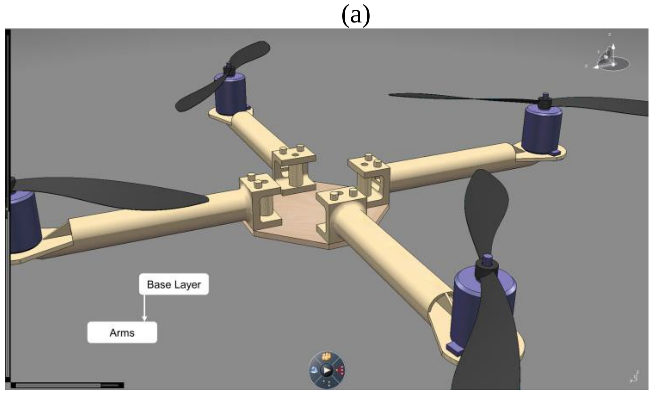
The model skeleton is a refined version of the one developed in architecture selection and includes more details relevant to the detailed vehicle geometry. Model skeletons serve two important functions. In the early stages of development, they assist in capturing the configuration layout of the platform. The model skeletons are continuously refined as development progresses, but defining high level layouts and subsystem divisions helps to organize the process.

The other role of model skeletons is to facilitate data flow down from top level design parameters to low level detailed geometry. The model skeleton contains all geometry that is referenced by more than one part. When it is time for a model update, first the skeleton changes to match the new values of top level design parameters. Then, individual parts reference the model skeleton for shared geometric features. Without model skeletons, updates to the 3-D model become iterative. Iterative operations on detailed 3-D models can be very slow. Operating one time on a model skeleton which is in turn referenced by all the other geometry is much faster. It is important to note that the model skeleton contains all the interface geometry between parts. Therefore, when the parts are manufactured interface geometry is more likely to be consistent.

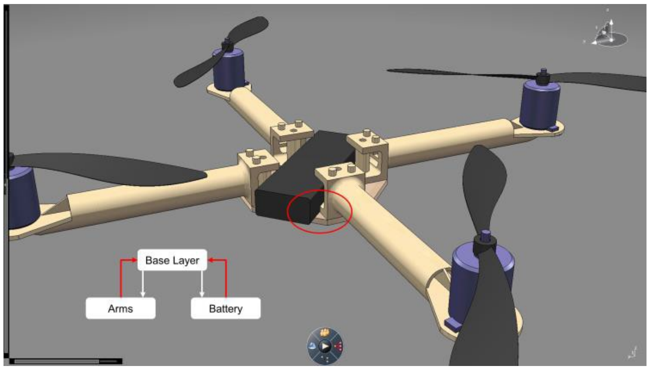
The role of the model skeleton in facilitating design data flow down is a crucial enabler of design automation as it enforces unidirectional dependencies. Without a model skeleton, the designer is forced to encode geometric references directly between features of separate components. This is especially true at component interfaces. The presence of such references inherently creates a global hierarchy of features that is exceedingly difficult to track. When a design parameter is changed, the geometry is updated in the order of the hierarchy. However, there is no mechanism to prevent circular dependencies between parts in this situation.

This idea is explained by the simple example depicted in Figure 17. The goal of the progression in Figure 17 is to place the lower chassis layer, the four multirotor arms, and the battery in space. In this example, there is no model skeleton. This notional 3-D model has been set up as follows: the lower chassis layer is arbitrarily placed in space (shown in Figure 17(a)). Then, the arms are placed by aligning each to the edges of the chassis layer (shown in Figure 17(b)). The battery is placed in the middle of the chassis layer according to design logic that requires the SUAS's center of gravity be in the center of the vehicle. However, the design parameters driving this design have resulted in interference between the base of the arms and the battery (shown in Figure 17(c)). An increase in the base layer width is necessary to fix this problem, but the location and shape of the battery and arms are required to compute the new width. Since the location of the battery and arms depends on the layer width, the only solution is to iteratively search using a guess and check method for the correct increase in width. The required number of this type of iteration grows quickly as more parts and features are added to the model. The result is a very computationally expensive model update cycle which makes automated changes infeasible.

(a)



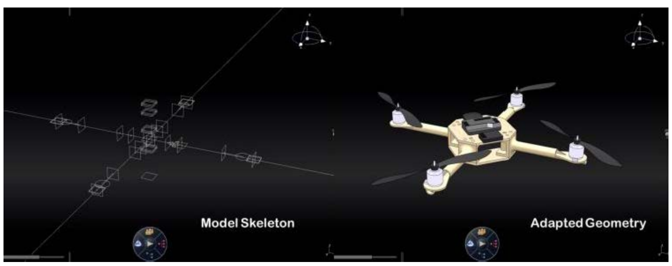
(b)



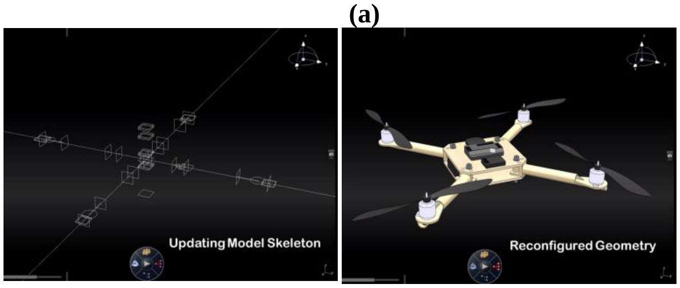
(c)

Figure 17. Example of a circular dependency between geometric features in the absence of a model skeleton.

Figure 18 shows how a model skeleton is used to eliminate circular dependencies. Top level design parameters drive the dimensions on the skeleton model, causing the variations in the skeletons of Figure 18 (a) and (b). Then all other components and geometric features reference the skeleton model for size and placement information. The result is an organized and unidirectional hierarchy of geometric feature dependencies.



(a)



(b)

Figure 18. Model updates using a model skeleton approach.

The physical model contains more than just 3-D representations of the SUAS. It is paired with logical rules which enforce design logic during model updates. These rules can be in the form of equations, conditional statements, checks, or loops. Logical rules are the mechanism by which impossible or invalid geometries are avoided.

The interactions between the model skeleton, logical rules (in the form of equations and checks), and lower level detailed geometry are best illustrated by an example. Consider one of the arms of the multirotor SUAS. The overall length of the multirotor arm and the spacing between the chassis layers are design parameters determined by the sizing algorithm. The model takes these parameters along with the names of the selected motor and propeller as inputs. It then uses equations to compute the dimensions of the arm's base, the diameter of the motor mount pad, and the location and size of the interface geometries. These computed dimensions are applied to the skeleton model which adjusts accordingly. Changes to the skeleton model are propagated to the detailed features, and then the model proceeds to invoke several rules and checks to enforce design logic. Several examples are

- A rule restricts any geometry from occupying the volume swept out by the propeller as it rotates.
- A check ensures all sharp corners are filleted so the arm is compatible with a 3-D printing process. If the check finds insufficiently rounded geometry, it applies fillets.
- The bending moment at the root of the arm's cantilevered tube varies with arm length. When the arm length is updated a check computes whether the thickness of the beam is enough to handle the new bending moment and increases the thickness if necessary.
- A check determines if the bolt pattern of the selected motor matches the holes on the motor mount pad. If they do not match, the holes are modified.
- A check evaluates the model to determine if there is any material obstructing the space directly below the motor bolt pattern. This space needs to be clear to permit an Allen key to install the motor mount bolts. If the check finds that the space is obstructed, it rotates the bolt pattern until it is clear.
- A conditional check is invoked if the user has requested a camera sensor on the SUAS. This check determines if a part of the arm is obstructing the field of view of the camera. If the field of view is obstructed, it flags other portions of the physical model so they can adjust accordingly.

<u>Manufacturing time modeling</u>

The majority of manufacturing time is contained in the time required to 3-D print components and prepare them for final assembly. A set of four arms for a quadcopter may take over 20 hours to print and an additional 8 hours remove and dissolve support material left over by the printing process. In order to accurately estimate the total manufacturing time, it is critical to estimate the time required to print parts.

Print time depends heavily on the specific 3-D printer used to fabricate parts. Accordingly, the 3-D printer model is left as an input for the user. For the work presented in this paper, fabrication takes place in a Stratasys® uPrintSE printer. Software provided with the 3-D printer provides

time estimates for a given print job. However, computation of these estimates requires the actual STL file for the part to be printed which is generally unavailable until the final stages of the automated design cycle. Instead, a regression model is used to estimate the print time for multirotor arms before a 3-D model is generated. A range of varying arm sizes was loaded into the uPrintSE software and printing time estimates were recorded. This information was used to generate a regression model which is used to estimate the print time of a multirotor arm given values of the arm's scalable parameters. Figure 19 shows the regression model plotted over ranges of arm length and arm base height which are two of the arm's scalable parameters.

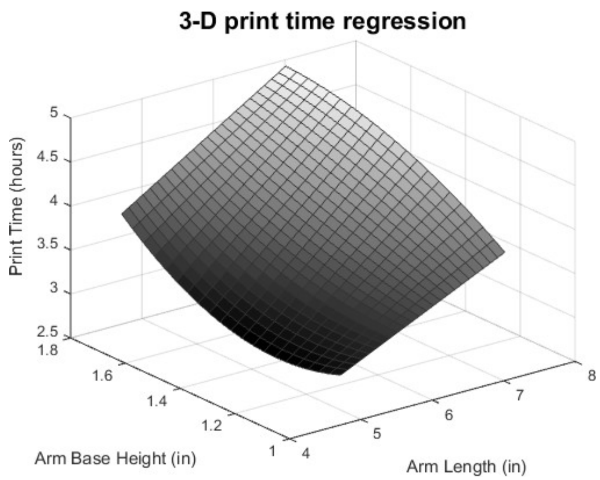


Figure 19. Depiction of the arm print time regression model with respect to two of the arm's scalable parameters.

RESULTS

The completed multirotor ADAPt design tool was used to generate proof-of-concept vehicles. The purpose of this exercise is to confirm two major objectives: (1) the design tool is able to generate a variety of vehicles suited for different missions and (2) the designed vehicle meets the input requirements and therefore can complete the intended design mission.

To accomplish this task, two notional missions and sets of mission and vehicle requirements were generated and the design tool was executed. The vehicles designed for the notional missions were fabricated and flight tests were performed to verify that mission requirements were met.

Bridge Inspection Mission

The first mission used to derive vehicle-level requirements is a notional bridge inspection mission. In this scenario, a Soldier needs to visually inspect the underside of a 600 ft. long bridge. The Soldier must inspect the surrounding areas and underneath the bridge supports which are spaced 33 inches apart. The SUAS should be very portable, so its weight is limited to 5 lbf. Listed below are the high level requirement inputs used to design the bridge inspection multirotor.

Mission requirements:

- Endurance: at least 10 min
- Maneuverability: High, with hover capability
- Maximum Size: 33 in.
- Maximum Weight: 5 lbf.
- Payload Capacity: 0 lbf.

Sensor Options:

- Live Video Feed

Material and Inventory Constraints:

- Depleted supply of motor type: RCTimer HP2820
- Limited material supply of 3D-printing plastic and plywood

Inputting these design requirements results in the design summarized in Table 7. The "Design" column in Table 7 lists the estimated values for each metric predicted by the design tools. A rendering of the physical that is generated is shown in Figure 20.

Table 7. Design requirements and resulting estimated design characteristics for the bridge inspection mission.

Requirement	Value	Design
Max Outer Dimension (in)	33.0	26.7
Max Weight (lbf)	5.0	2.98
Min Endurance (min)	10.0	11.2
Max Build Time (hrs)	22.0	16.1
Extra Payload (lbf)	0.0	0.99
Sensor	GoPro®	

Figure 20. Render of the physical model for the bridge inspection mission multirotor design.

Communications Relay Mission

A notional communications relay mission was selected for the second proof-of-concept vehicle. The communications relay mission requires a vehicle capable of hovering steadily with a heavy payload. While the bridge inspection mission emphasizes endurance, this mission's requirements emphasize payload in order to show the range of vehicles that the ADAPt Design tool is able to generate. The requirements for the communications relay mission are listed below:

Mission Requirements:

- Endurance: at least 7 minutes
- Maneuverability: Normal, with hover capability
- Maximum Size: 50 in.
- Maximum Weight: 20 lbf
- Payload Capacity: 4 lbf

Sensor Options:

- None

Material and Inventory Constraints:

- None

Inputting these design requirements yields the design summarized in Table 8. A rendering of the resulting physical model is shown in Figure 21.

Table 8. Design requirements and resulting estimated design characteristics for the communications relay mission.

Requirement	Value	Design
Max Outer Dimension (in)	50.0	34.1
Max Weight (lbf)	20.0	5.16
Min Endurance (min)	7.0	7.18
Max Build Time (hrs)	40.0	33.1
Extra Payload (lbf)	4.0	8.63
Sensor	none	

Figure 21. Render of the physical model for the communications relay mission multirotor design.

A comparison of the designed vehicle results shows the how the ADAPt design tool is able to select the best feasible vehicle alternative for a given mission. For the communications relay mission, a six-armed multirotor is generated for high payload capacity. Comparing to the bridge inspection mission, a quadcopter is generated with a three-layer chassis to minimize vehicle size while providing space to attach the video camera.

Requirements verification

While many of the physical characteristics such as vehicle weight and maximum dimension are relatively easy to verify, performance characteristics such as payload capacity and flight endurance require flight testing to confirm. The bridge inspection mission multirotor was therefore fabricated and flight tests were conducted to verify the design met the endurance and maneuverability requirement. The maneuverability requirement was verified qualitatively by performing rapid accelerations starting from a hover. The vehicle performing this maneuver is show in Figure 22.

Figure 22. Bridge inspection multirotor with camera feed performing an acceleration maneuver.

Hover endurance was tested as a part of the flight tests. The SUAS was flown with a fully-charged battery. The position of the vehicle was controlled manually, and the pilot attempted to maintain a steady position and altitude. Once the battery voltage decreased to a level which indicated low capacity, the flight test concluded. An ending battery voltage of 3.5V was determined using the minimum safe voltage limit of LiPo batteries. For each flight test effort, battery voltage at the start and end were recorded as well as the flight duration. Post flight test, the batteries were recharged and the electric charge input was recorded. This value was used as an estimate for electric charge consumed during flight. Table 9 gives the endurance times recorded. As seen from the results, the energy and force model in conjunction with the mission model underestimates the endurance of the vehicle when compared to the actual flight test results.

Table 9. Bridge Inspection Quadcopter Endurance Flight Test Results.

Test #	Actual Endurance (min)	Predicted Endurance (min)
1	14.8	11.2
2	15.4	11.2
3	15.0	11.2

Several sources of uncertainty exist and may be key contributors to the discrepancy between predicted and actual endurance times. The battery capacity used to calculate the predicted endurance is based on the nominal battery capacity. During the flight test, the actual energy drawn from the battery varies with environmental conditions and pilot input. Furthermore, the mission model used to predict the vehicle endurance does not include mission segments. Inclusion of varying mission segments which accurately

represent different phases of flight into the model can help reduce error in the endurance estimation.

CONCLUSIONS

The real-time evolution of U.S. Army operations is fast paced and requires Soldiers to make quick decisions. SUAS have recently proven effective in providing Soldiers the intelligence they need to make these decisions. This paper presents an approach to equipping Soldiers with custom tailored SUAS to meet these rapidly changing and unpredictable mission needs using a design on-demand philosophy. ADAPt Design is a method created to enable an on-demand by extending the notion of product family design.

This paper demonstrates ADAPt Design's utility in automating the design and manufacture of a multirotor SUAS, which serves as the initial test case of the methodology. The end result is an integrated set of executable design tools, which were used to design two vehicles. The fact that dissimilar designs were the outcomes of contrasting mission requirements serves as an initial validation of the method.

However, the performance estimates obtained from the modeling tools exhibit deviations from the actual measured performance. This demonstrates that the method is limited by fidelity and amount of uncertainty in the models. Furthermore, multirotor SUAS are relatively simple systems. Application of the ADAPt Design method to more complicated systems would require more rigorous development and modeling techniques. Despite these limitations, the authors believe that, with appropriate modeling tools, the ADAPt Design method is extensible to other products and systems.

Author contact: Arthur Cheng artcheng@gatech.edu
Zachary Fisher zcfisher@gatech.edu
Raphaël Gautier raphael.gautier@gatech.edu
Nathan Beals nathan.e.beals2.ctr@mail.mil
K. Daniel Cooksey daniel.cooksey@aerospace.gatech.edu
Dimitri Mavris dimitri.mavris@aerospace.gatech.edu

REFERENCES

[1]"The U.S. Army Operating Concept; Win in a Complex World," Fort Eustis, VA, 2014, TRADOC 525-3-1.

[2]Johnson, D. E., Markel, M. W., and Shannon, B., *The 2008 Battle of Sadr City: Reimagining Urban Combat*, RAND Corporation, Santa Monica, CA, 2013.

[3]"Eyes of the Army–US Army Roadmap for Unmanned Aircraft Systems 2010 - 2035," Ft. Rucker, AL, 2010.

[4]"Army Unmanned Aircraft System Operations," Washington, DC, 2006.

[5]Piper, S. S. R., "Small UAV Provides Eyes in the Sky for Battalions," *Army News Service*, 2005.

[6]Rapid Equipping Force, Online, Accessed 21 January 2015.

[7]Fisher, Z., Locascio, D., Cooksey, K. D., Mavris, D., and Spero, E. (in press), "ADAPt Design: a methodology for enabling modular design for mission specific SUAS," *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 2016 ASME*, American Society of Mechanical Engineers, 2016.

[8]Forsberg, K. and Mooz, H., "System Engineering for Faster, Cheaper, Better," *INCOSE International Symposium*, Vol. 9, No. 1, 1999, pp. 924–932.

[9]Mangum, P., Fisher, Z., Cooksey, K. D., Mavris, D., Spero, E., and Gerdes, J. W., "An Automated Approach to the Design of Small Aerial Systems Using Rapid Manufacturing," *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 2015 ASME*, American Society of Mechanical Engineers, 2015, p. V02BT03A042.

[10](Roger) Jiao, J., Simpson, T. W., and Siddique, Z., "Product family design and platform-based product development: a state-of-the-art review," *Journal of Intelligent Manufacturing*, Vol. 18, No. 1, 2007, pp. 5–29.

[11]Simpson, T. W., Maier, J. R., and Mistree, F., "Product platform design: method and application," *Research in Engineering Design*, Vol. 13, No. 1, 2001, pp. 2–22.

[12]Sawhney, M. S., "Leveraged high-variety strategies: From portfolio thinking to platform thinking," *Journal of the Academy of Marketing Science*, Vol. 26, No. 1, 2007, pp. 54–61.

[13]Li, L., Huang, G., and Newman, S. T., "Interweaving genetic programming and genetic algorithm for structural and parametric optimization in adaptive platform product customization," *Robotics and Computer-Integrated Manufacturing*, Vol. 23, No. 6, 2007, pp. 650 – 658, 16th International Conference on Flexible Automation and Intelligent Manufacturing.

[14]Ulrich, K., *Management of Design; Engineering and Management Perspectives*, chap. Fundamentals of Product Modularity, Springer Netherlands, 1994, pp. 219–231.

[15]Ulrich, K. and Eppinger, S. D., *Product Design and Development*, McGraw-Hill, 1995.

[16]Pirmoradi, Z., Wang, G. G., and Simpson, T. W., *Advances in Product Family and Product Platform Design*,

chap. A Review of Recent Literature in Product Family Design and Platform-Based Product Development, Springer Science and Business Media, New York, NY, 2007, pp. 1–46.

[17]Locascio, D., Cooksey, K. D., Mavris, D., and Spero, E., (in press) "A Framework for Integrated Analysis, Design, and Rapid Prototyping of Small Unmanned Airplanes," *AIAA Aviation 2016*, American Institute of Aeronautics and Astronautics.

[18]Hwang, C.-L. and Yoon, K., *Multiple Attribute Decision Making*, Vol. 186, Springer-Verlag Berlin Heidelberg, 1981.

A Framework for Integrated Analysis, Design, and Rapid Prototyping of small Unmanned Airplanes

David B. Locascio[1], Coline L. Ramee[2], K. D. Cooksey[3], and Dimitri N. Mavris[4]
Georgia Institute of Technology, Atlanta, Georgia, 30332-0150, United States

A vital requirement of the modern combat environment is to gain and maintain situational awareness to facilitate effective squad-level decision making. This paper presents a part of the research undertaken by Georgia Institute of Technology (Georgia Tech) in collaboration with the Army Research Laboratory (ARL) in developing design capabilities for small unmanned aerial systems (sUAS). As part of this effort the team developed a toolset capable of creating mission-specific fixed wing aircraft assets that can be rapidly tailored and manufactured at a forward operating base. The toolset includes a physics-based analysis model to generate feasible aircraft designs from a family of designs, a decision making tool to select the optimal design for a mission, and a parametric CAD model. The CAD model accepts sizing parameters from the design algorithm and uses them to scale baseline part files, which can then be used to rapidly manufacture vehicle parts. Several sets of mission requirements were chosen, leading to unique fixed wing aircraft designs which were manufactured and flown. The process described herein can be used to develop and fabricate small unmanned airplane designs to fulfill rapidly changing squad-level mission-specific operational needs, but can also be applied to other vehicle architectures.

I. Introduction

Current U.S. Army Unmanned Aerial Systems (UAS) are used for support of tactical operations via the gathering of intelligence, surveillance, and reconnaissance (ISR). Ideally, UAS assets can be deployed by troops on-demand to acquire intelligence in real-time. RAND Corporation conducted an assessment of U.S. military operations in Baghdad, Iraq and concluded that modern combat requires decentralized decision making, stating that

> "the enemy is fleeting, which means that decentralized decision making is required. Units at the brigade level and below must therefore have access to the information and other capabilities required to support the rapid decisions necessary to deal with a highly mobile enemy … and to enable effective, independent action" [1].

The U.S. Army roadmap for UAS between 2010 and 2035 further supports this conclusion, stating

> "UAS require and enable accelerated multi-echelon, decentralized decision-making, and execution, significantly changing the tempo and dynamics of operations. Lower echelon leadership must be empowered with authority and bandwidth to employ UAS as their changing situation dictates, operating at a tempo that is faster than higher echelon leadership can affect" [2].

Small unmanned aerial systems (sUAS) have increasingly been used to provide Actionable Intelligence in order to facilitate decentralized decision making. These systems can perform ISR, security, manned-unmanned teaming,

---

[1] Graduate Research Assistant, School of Aerospace Engineering, 270 Ferst Dr, Mail Stop 0150, AIAA Student Member
[2] Graduate Research Assistant, School of Aerospace Engineering, 270 Ferst Dr, Mail Stop 0150
[3] Research Engineer, Aerospace Systems Design Laboratory, 275 Ferst Drive #3, Atlanta, GA 30332, AIAA Vice-Chairman Altanta Section.
[4] Regents Professor, Boeing Professor of Advanced Aerospace Analysis, Director of the Aerospace Systems Design Laboratory, School of Aerospace Engineering, 270 Ferst Dr, Mail Stop 0150, AIAA Fellow.

communications relay, IED searches, identify enemy combatants, and perform advance scouting to reduce risk to the Soldier [3] [4].

While the overall objective of the ARL-Georgia Tech collaboration is to improve situational awareness and effectiveness of squads via sUAS, squad requirements span a broad range relative to the size of the vehicle. A soldier may require a sensor package that can be utilized to map a building with a sensor of only a few grams and a few minutes of endurance with upper constraints on the vehicle size due to door and corridor size. On the other end of the spectrum, a soldier may want a system capable of surveillance for an hour or more with no size limitations. Mission requirements evolve on a day-to-day basis, and may not be foreseen at deployment. At the sUAS scale, asset designs are highly sensitive to mission requirements.

Three approaches can be employed when trying to develop an asset that best satisfies diverse soldier needs.

- *Multi-mission asset*: One UAS is generated, which while able to cover all mission needs sacrifices performance on some of the possible missions.
- *Set of optimized assets*: A set of optimized vehicles is created which each can perform one or more missions very well, but require troops to carry an overwhelming number of vehicles to account for all mission possibilities.
- *Asset On-Demand*: One-off asset which is specifically tailored and optimized to perform one mission as per some input criteria .

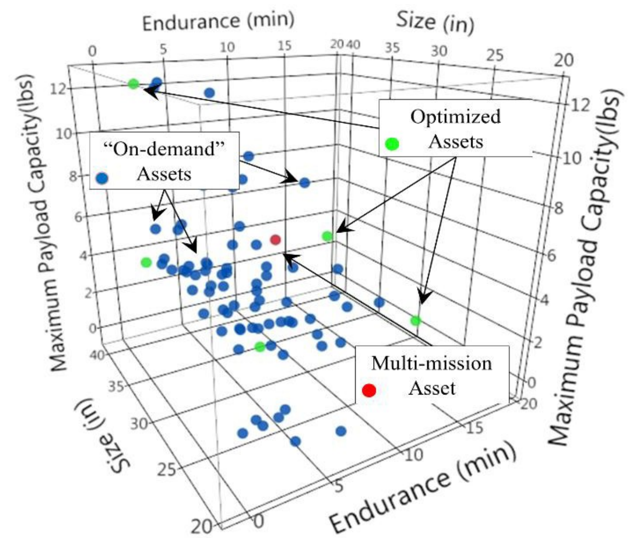


Figure 1: Asset Class and Addressing Diverse Mission Needs

While traditionally there have been technical hurdles limiting the on-demand approach, access to rapid manufacturing and rapid engineering tools and equipment have helped to eliminate these boundaries. This allows for a soldier to access an improved space of solutions, but there are some tools required to allow for forward deployed soldiers to perform rapid engineering and manufacturing of specialized sUAS systems.

Previous work includes a multidisciplinary framework built on simulataneous application of decomposition and re-composition, implemented in order to establish a structured and traceable method to evaluate mission effectiveness of microsystems [5]. This led to the development of the Interactive Reconfigurable Matrix of Alternatives, which assists with comprehending the large concept solution space. Fundamental mission requirements included endurance, adaptability, path planning, and communications [6].

This paper seeks to build off of previous tools created by Georiga Tech and ARL by Mangum [7] and develop a process by which a soldier can input requirements and then assemble a vehicle for operation. The work outlined in this paper follows the Aggregate Derivative Approach to Product Design (ADAPt) methodology as described by Fisher [8]. A similar use of ADAPt Design for on-demand multirotor design and fabrication is presented in another paper by Cheng et. al [9].

II. Research Objectives

The objective of the current work is to develop an engineering process to provide a soldier with a method of inputting their needs, such as mission requirements, and returning to the soldier a small unmanned fixed wing airplane design tailored for those needs. To support this, there is a physics based performance and analysis model coupled with an easy-to-build fixed wing aircraft design, modeled in a CAD environment. The baseline design is scaled and combined with commercial off-the-shelf parts based on the output of the analysis and sizing model.

The on-demand process can be easily explained with an analogy to Lego® in Figure 2. Lego® bricks are made up of modular pieces which can be assembled into different models. The user is provided with instructions to enable different systems to be built out of the same components. For this work, a shoebox-sized box of off-the-shelf components must be included, containing parts that cannot be easily manufactured on-site – motors, propellers, batteries, and other electronics. These parts are commonly used between designs and are combined with modular parts in order to build the required system.

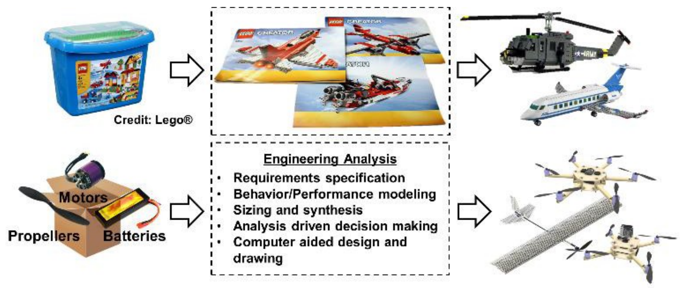


Figure 2: Design On-Demand illustrated via Analogy to Lego® [9]

This paper describes the development of the tools and models necessary to enable on-demand design for a small fixed-wing aircraft.

III. Approach

Figure 3 shows the five elements used to enable rapid automated design and manufacturing.

1. A component database, containing the technical specifications and size of several off-the-shelf components
2. A multi-disciplinary performance analysis program, integrating physics-based and data-driven analysis
3. A multi-criteria decision making environment to select the optimal design from a Pareto frontier for specific mission requirements
4. A parametric CAD model
5. Rapid prototyping tools

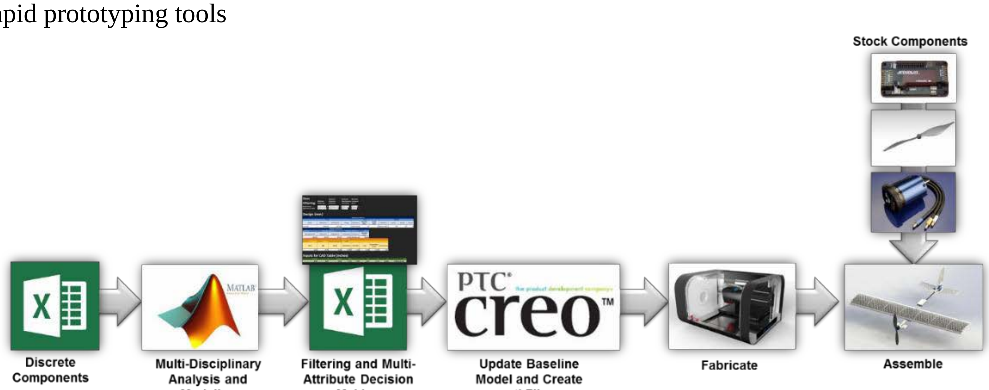


Figure 3. Modeling, Design and Prototyping Process

A. Component Database

Since some components such as motors and other electronics cannot be manufactured easily using rapid manufacturing techniques, a stock of commercial off-the-shelf (COTS) parts is maintained. In order to model these COTS parts within the multi-disciplinary performance analysis program, a database of their geometries and other relevant technical specifications is created. This includes a selection of motors, propellers, and batteries. Fixed

components include the flight controller, GPS, servo motors, and electronic speed controllers (ESCs). The component database was created with Excel, and contains the characteristics of all the off-the-shelf components that will be considered for design solutions, such as weight and performance characteristics. These parts are all small and interchangeable between designs. The database also contains airfoil profiles that will be considered in the design. Data was gathered for the component database using test data from literature and manufacturer specifications. Propeller data was mostly obtained from the University of Illinios at Urbana-Champaign (UIUC) Propeller Database which contains many hobby-type propellers. The performance of a propeller is given by the efficiency as a function of the advance ratio at different rotational speeds as shown in Figure 4. Small aircraft operate at low advance ratios, since the free stream velocity is low compared to the rotational speed of the propeller. For low advance ratios the performance at different RPM has small variance, so a least square fit is performed on the points to obtain a single performance curve. The advanced ratio is given by $J = \frac{V_\infty}{nD}$ where V_∞ is the airstream velocity, n is the number of rotations per seconds and D is the propeller diameter. Propeller diameter is stored in the database, the RPM is an optimization variable and the aircraft cruise speed is V_∞, the free stream velocity. The resulting efficiency η is the ratio of useful power output over input power from the motor.

The goal when selecting components for the library was to create a large design space via a variety of components which are used in the physics-based performance model. By testing the toolset for a variety of missions, components that are rarely used can be eliminated to reduce the number of discrete components that must be contained in the shoebox.

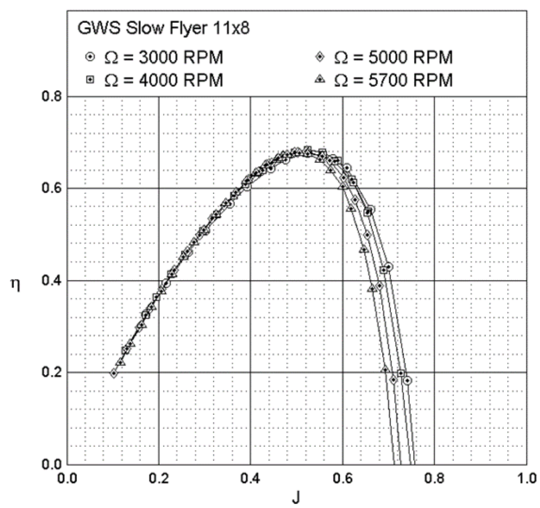


Figure 4. GWS Slow Flyer 11x8 Propeller [10]

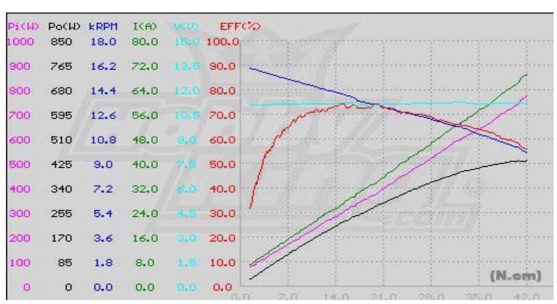


Figure 5. NTM Dyno Test Data [11]

To model the motors a very high level model is used. The behavior of each motor is characterized in the database by a maximum current, a K_V and a torque/RPM at different voltage. The K_V value represents the maximum RPM per volt applied on the motor without load, this value is called RPM_0. The modeled relationships between applied voltage, V, current, I, and motor RPM are given by Equations 1, 2 and 3, where τ is the motor torque and a is the best fit slope of the torque/RPM curve

$$RPM_0 = K_V * V \tag{1}$$

$$RPM = a * \tau + RPM_0 \tag{2}$$

$$I = \frac{\tau}{30}\pi K_V \tag{3}$$

A variety of motors of different sizes was selected, accepting various propeller and voltages, with available test data in order to populate a wider design space.

The number of discrete components and the range they span are listed in Table 1. Very little data is available for RC sized propellers, which led to a sparsely populated design space. A variety of 3S and 4S LiPO batteries are selected and contain weight, capacity, discharge rate, and dimensional data. The motors selected are appropriate voltages for the motors selected. Three airfoils are selected – a highly cambered Sellig 1223, a symmetric NACA 0012, and a moderately cambered Clark Y airfoil in order to populate a wider area of the design space.

Table 1. Component Library

Component	Number of Options	Minimum		Maximum
Motors	5	28x32 mm		35x44 mm
Propellers	2	9x7.5 inches		11x8 inches
Batteries	9	3 cells, 1,300 mAh		4 cells, 8,000 mAh
Airfoils	3	Sellig 1223	Clark Y	NACA 0012

B. Multi-Disciplinary Analysis and Modeling

The analysis input parameters are discrete components, performance varables such as maximum acceleration or rate of climb, geometry parameters, and RPM. The design is feasible if no component constraints are violated (such as current overdraw) and if the objectives are met. The feasibility of the propulsion system (battery, motor, rotation speed) can be analyzed separately from the rest of the variables. Because there is a small number of each component in the current database a full factorial on these three variables can be performed. Sets which result in a current draw that is too high for the battery or the motor, as well as sets resulting in an endurance smaller than the constraint, are discarded before the optimization algorithm is run. Generating feasible propulsion sets decreases the number of variables in the NSGA-II optimization and improves the running time.

Performance requirements are represented as constraints, and allow for feasible ranges, preventing any designs outside of those ranges. This includes the parameters and ranges in Table 2.

Table 2. Parameters and Their Feasible Ranges

Parameter	Type	Range
Propeller	Discrete	N/A
Propulsion Set	Discrete	N/A
Airfoil	Discrete	N/A
Span	Continuous	.5 to 1.53 meters
Chord	Continuous	.15 to max printer build dimension (meters)
Payload Mass	Continuous	.05 to .5 kg
Speed	Continuous	6 to 30 m/s
Rate of Climb	Continuous	.5 to 5 m/s
Load Factor	Continuous	1.05 to 2
Acceleration	Continuous	.5 to 5 m/s^2
Launch Speed	Continuous	4 to 8 m/s

This further decreases the size of the design space and helps to ensure feasible, realistic designs. Launch speed is constrained to 8 m/s since a hand launch is desired. One important constraint on the problem is the printer area – this limits how large the wing chord can be, and forces printing the wing in multiple sections. The span is limited to 1.53 meters in order to ensure the vehicle is easy to assemble and reduce wing twist and increase ease of assembly. This length was selected to allow for a maximum of nine wing sections for the 3-D printer used.

Multi-objective optimization is performed using an NSGA-II algorithm by varying the above parameters in order to generate solutions. The parameters for optimization are to minimize print time and launch speed while maximizing endurance, payload, speed, rate of climb, load factor, and acceleration. A Pareto frontier is created and output to an Excel spreadsheet with performance characteristics and the required CAD dimensions and parameters for each design.

An overview of the analysis is given in Figure 6. Continuous variables and discrete variables from the database are fed to the analysis. First the mass, print time, propulsion system performance and aerodynamic parameters are estimated. Then the constraints on each flight segment are analyzed based on the aircraft wing loading and power loading. If the vehicle is able to perform the mission described in the input "Mission Description" the design is feasible. The objective is to minimize manufacturing time, maximize endurance, payload mass and the performance parameter (acceleration, cruise speed, rate of climb,…) of the feasible designs. By creating a Pareto frontier of solutions, those solutions can be used indefinitely without needing to rerun the analysis model, and cover every possible usecase.

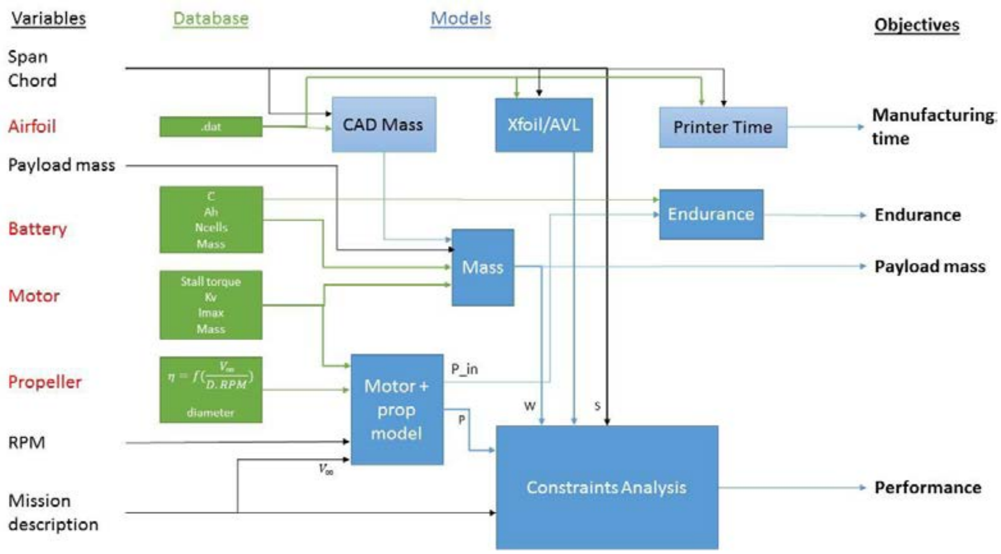


Figure 6. Modeling Overview

C. Filtering and Multi-Attribute Decision Making

Since the Pareto frontier is a set of non-dominated solutions across the entirety of the design space, the solutions must be compared via some multi-criteria decision-making technique. To select the best solution for a mission, an Excel-based decision making environment was created, where a user can filter solutions and enter weightings to get the optimal solution for a particular set of requirements for a mission. TOPSIS was used in order to rank and select from the Pareto solutions. The parameters selected for user input are payload, manufacturing time, endurance, top speed, and acceleration.

Two missions with unique requirements were selected as test cases. Appropriate aircraft for each mission were built and flown to verify the model and overall process, and are depicted in Table 3. These were chosen to maximize the variety of the designs and explore the limits of the model and process.

Table 3. Missions Selected for Fixed Wing Design Study

Mission	Description
Reconnaissance	Long endurance surveillance – requires some minimum payload, want maximum flight time on target
Hot Payload Delivery	Deliver a payload as fast as possible – lower bound on payload and endurance while minimizing manufacturing time and maximizing flight speed

These two solutions are pictured in Figure 7 in a three-dimensional design space of endurance, build time, and payload, along with the other Pareto solutions.

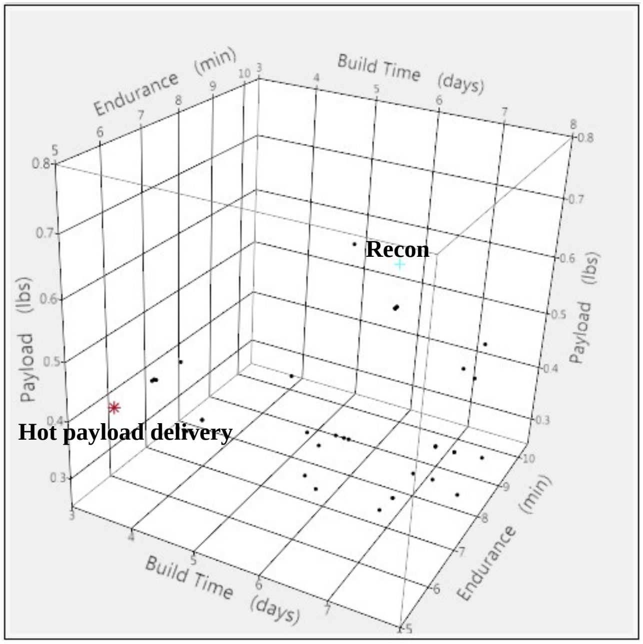


Figure 7: 3-D Design Space with Selected Designs

D. Baseline CAD Model

The baseline CAD model is actually the first tool that is created, since the analysis model is dependent on regressions which require a family of aircraft designs. The CAD models must be prepared to accept a range of inputs via parameterization, where dimensions and components are driven by equations and variables. After this is setup, regressions are created by examining the print time and mass of various sizes of the different parts.

In the work flow of generating a mission-specific vehicle, the selected design solution parameters are input from the Excel tool into the parametric CAD environment, which updates the baseline design's component selection and part geometries such as airfoil type, span, chord, battery, motor, and propeller selection. Finally, the designed parts are saved as .STL files and fabricated using a 3-D printer. There are additional constraints inherent in the fabrication process that must be taken into account for the performance analysis. As mentioned earlier, the 3-D printer's limited build area constrains the aircraft wing so that it must be printed in multiple sections. These manufacturing constraints impact model parameters such as weight and allowable chord length. Since ABS is much denser than EPP foam (a typical material for aircraft of this size), minimizing weight was a dominant design factor for the baseline design. The internal structure of the wing section was designed as a honeycomb fill to provide a high amount of bending and torsional stiffness while still keeping the weight low [12]. The baseline CAD model is depicted in Figure 8.


Figure 8. Fixed Wing Baseline CAD Model

The ADAPt process was implemented in the vehicle design in order to develop common interfaces for components. One example of enabling modular design is given in

Figure 9, which depicts the interface between the motor and the main shaft of the aircraft. The adapter plate (middle) can be configured to different motor diameters and screw sizes and spacing. Another example is the battery cage, which maintains a common interface of the main fuselage shaft while the part itself changes based on the battery

selection.

Figure 9: Exploded view of fixed wing motor mount assembly

Two goals of an aircraft design for rapid design and production are modularity and ease of assembly. The vehicle was designed around a single main shaft of carbon fiber. All components slide easily onto the shaft and lock to one another through alternating teeth, and several components are fixed to the shaft through the use of bolts. The wing is printed in sections since the build area of the printer used was only 6 inches by 8 inches. In order to have a 3-D printed wing that was both light and strong, a honeycomb fill was designed into each wing section. The wing is assembled using two carbon fiber spars and packing tape wrapped around to form the skin.

The components slide onto the shaft in a specific order and interlock with alternating teeth in order to prvent rotation around the shaft. The empennage is an all-moving tail, directly mounted onto the servo motors. It is important to note that there are no ailerons on the wing, so all roll control comes from the tail. This reduces design complexity and the number of servo motors needed, but small metal geared servos are often easy to strip.

IV. Conclusion

Several designs were built and flown to ensure operation of the aircraft. Figure 10 shows the two selected design cases and the changes between them, and Figure 11 shows a fully assembled aircraft that underwent test flights. One issue is the lengthy print time of 7 or 9 wing sections, each taking over 8 hours to print, with the middle section reaching a hefty 13 hours. Since the goal of this work is to have an asset available within 24 hours, this becomes a showstopper with current 3-D printing technology. However, there is no doubt that 3-D printing technology will improve, and it may be possible to improve the baseline design in order to reduce the wing's print time.

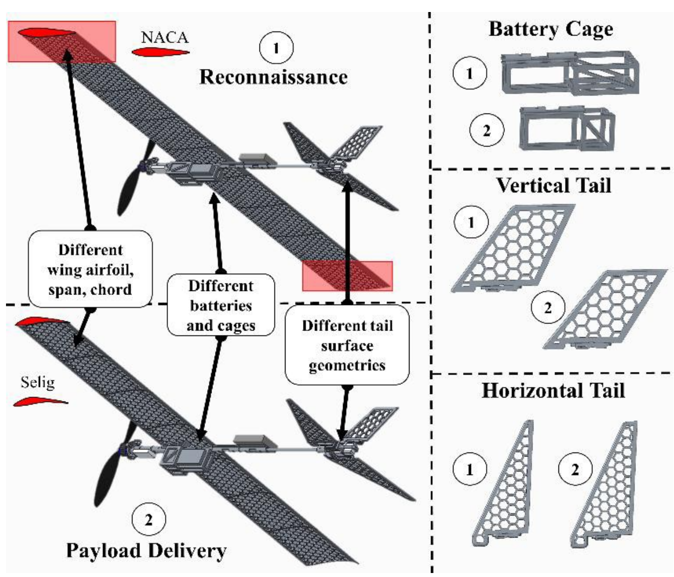


Figure 10. Depiction of a fixed wing model update as a result of a change in requirements

Figure 11. Fixed wing design model

Modern-day military operations require increasingly decentralized decision making, and SUAS can be employed to provide intelligence to soldiers. While multi-mission assets face degradation of performance across a mission envelope and are not always available to the Soldier, on-demand assets are now possible due to improvements in scaled-down rapid manufacturing technology. In order to facilitate decentralized decision making, a unique toolset was developed in order to enable on-demand, tailored design and fabrication of a fixed wing aircraft. The process presented in this paper can be extended for other systems or sub-systems, scaled-up or scaled-down, by linking engineering analysis models to a parametric CAD model and employing rapid prototyping to quickly generate new designs on-demand.

References

[1] D. E. Johnson, M. W. Markel and B. Shannon, "The 2008 Battle of Sadr City: Reimagining Urban Combat," RAND Corporation, Santa Monica, CA.

[2] "U.S. Army Roadmap for Unmanned Aerial Sysesms 2010-2035," U.S. Army UAS Center of Excellence, Fort Rucker, AL., 2010.

[3] "Army Unmanned Aircraft Systems Operations," Department of the Army, Washington, D.C., 2006.

[4] S. S. R. Piper, "Small UAV Provides Eyes in the Sky for Battalions," Army News Service, 2005.

[5] Z. T. Mian, "A Multidisciplinary Framework for Mission Effectiveness Quantification and Assessment of Micro Autonomous Systems and Technologies," in *Dissertation, Georgia Institute of Technology*, Atlanta, GA, 2011.

[6] Z. T. Mian, P. Dees, L. Hall and D. Mavris, "Development and IMplementation of Micro Autonomous Systems and Technologies (MAST) Interactive Reconfigurable Matrix of Alternatives," in *Procedia Computer Science, Vol. 16, 213, pp. 708-717*, 2013.

[7] J. P. Mangum, Z. Fisher, K. D. Cooksey, D. Mavris, E. Spero and J. W. Gerdes, "An Automated Approach to the Design of Small Aerial Systems Using Rapid Manufacturing," in *ASME 2015 International Design Engineering Technical Conferences*, Boston, 2015.

[8] Z. C. Fisher, D. Locascio, K. D. Cooksey, D. N. Mavris and E. Spero, "ADAPt Design: a methodology for enabling modular design for mission specific sUAS," in *ASME IDETC (to be published)*, Charlotte, NC, 2016.

[9] A. Cheng, Z. Fisher, R. Gautier, K. D. Cooksey, D. Mavris and E. Spero, "A Model-Based Approach to the Automated Design of Micro-Autonomous Multirotor Vehicle Systems," in *AHS Forum 72 (to be published)*, 2016.

[10] C. B. Ananda, "UICC Applied Aerodynamics Group," 3 February 2015. [Online]. Available: http://m-selig.ae.illinois.edu/props/propDB.html.

[11] "NTM Dyno Test Data 3S LIPO," 2015. [Online]. Available: http://www.hobbyking.com/hobbyking/store/_16092_NTM_Prop_Drive_Series_35_42A_1250Kv_600W.html.

[12] L. Wahl, S. Maas, D. Waldmann, A. Zurbes and P. Freres, "Shear stresses in honeycomb sandwich plates: Analytical solution, finite element method and experimental verification," *Journal of Sandwich Structures and Materials*, vol. 14, no. 4, pp. 449-468, 2012.

[13] F. Samarrai, "University of Virginia Creates a 3D printed UAV Drone for the Department of Defense," 2014. [Online]. Available: http://additivemanufacturing.com/2014/09/25/university-of-virginia-creates-a-3d-printed-uav-drone-for-the-department-of-defense/. [Accessed 5th July 2015].